ABSTRACT

Title of Thesis: SOLVING MULTI-SCHOOL BUS ROUTING AND SCHEDULING PROBLEM

Zhongxiang Wang
Master of Science
2017

Thesis Directed By: Professor Ali Haghani
Department of Civil and Environmental Engineering

ABSTRACT:

School bus routing and scheduling is of huge importance in school transportation system operations. It is usually treated as two separated problems and is solved sequentially. But it is shown that such separation will lead to a worse solution than solving them together with respect to the number of buses and travel time. The rationale behind it and the key point connecting routing and scheduling problem – trip compatibility – is thus deeply studied. A Mixed Integer Programming model is proposed along with a School Decomposition Algorithm. The model and algorithm are tested on eight sets of randomly-generated mid-size problems in comparison to the existing models. The results show that the proposed model and algorithm can find a better solution using up to 30% fewer buses than the best traditional models in a reasonable amount of time.

SOLVING MULTI-SCHOOL BUS ROUTING AND SCHEDULING PROBLEM

by

Zhongxiang Wang

Thesis submitted to the Faculty of the Graduate School of the
University of Maryland, College Park, in partial fulfillment
of the requirements for the degree of
Master of Science
2017

Advisory Committee:
Professor Ali Haghani, Chair
Professor Cinzia Cirillo
Professor Paul Schonfeld

Acknowledgements

Since joining the Civil and Environmental Department of University of Maryland in 2015, I have received tremendous generous and life-changing help and instructions from my advisor, course professors and friends. With the help of all these, I have learned and grown more I could ever imgain. Here, I have to thank all of them for not only helping me to complete my master program but more importanly for helping me become a better person.

First, I would like to thank my advisor Dr. Haghani. He is the first one who led me to discover the beauty of Operations Research (OR) and ignited my passion for research. OR then becomes my research interest and the most important tool to finish this thesis. The chance he provided me – the exposure to the real world problem, the school bus routing and scheduling problem – let me learn the hardest from the practice. He is not that kind of advisor who pushs students day to day. But every opinion he gave, every edition he made and every discussion we had showed me the direction I should keep learning. He is my mentor who helped me pursue my academic dream. All he has taught me is more than enough I need to finish my master program. I am honored to join Dr. Haghani’s group.

Also, I need to thank all professors I had courses with: Professor Paul Schonfeld， Professor Cinzia Cirillo, Professor Lei Zhang, Professor Gang-Len Chang, Professor Barton Forman, Professor Michael Ball and Professor Michael Fu. Their profound knowledge at different research field broadened my horizon and let me see the various possiblilities of researches as well as the useful tools can be used for my own research. All of those, although may not become my major research interest, form the foundation of my knowledge and help me at all aspects of research.

I also appreciate all the help I received from my friends, the friends in Dr. Haghani's group, the friends I cooperated with to finish course projects and all the friends who inspired me. Entering graduate school is terrifying, doing research is sometime boring and putting all the efforts but achieving little is depressing. I cannot think of myself conquering all the difficulties without the help and encouragement from my friends.

There are a few special names I have to mention. First, Dr. Ali Shafahi who taught me a lot about the school bus routing and scheduling program from scratch. I have learned a lot from the projects we worked together. Also, I would like to thank Miss. Yeming Hao, who is my best friend and also the president of Intelligent Transportation Society and Institute of Transportation Engineers University of Maryland Student Chapter. She helped me with every pieces of my live here.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Problem Classifications:

S: Single-school

M: Multi-school

HO: Homogeneous Fleet

HT: Heterogeneous Fleet

OVRP: Open Vehicle Routing Problem

SBR: School Bus Routing

SBRS: School Bus Routing and Scheduling

SBS: School Bus Scheduling

TSP: Traveling Salesman Problem

m-TSP: Traveling Salesman Problem with m Trips

VRP: Vehicle Routing Problem

Objectives:

BAL: Balance Between Trips

DC: Depot Cost

ELP: Early and Late Penalty

MinB: Minimizing Number of Buses

MinB+TT: Minimizing Number of Buses While Minimizing Total Travel Time

MinN: Minimizing Number of Trips

MinTT: Minimizing Total Travel Time

NOB: Number of Buses

NOT: Number of Trips

PLT: Penalty (generalized penalty other than specific ELP)

SWD: Student Waiting Time

TST: Total Service Time (Total Travel Distance + Vertex Service Time)

TTD: Total Travel Distance

Constraints:

C: Capacity Constraint

CB: Chain Barring Constraint

LOG: Logistic Constraint

MNS: Minimal Number of Student to Create a Bus Trip

MRT: Maximum Ride Time

SBL: Sub-tour Elimination Constraint

TW: Time Window Constraint

Algorithms:

ARL: Allocation-Routing-Location

B&B: Branch and Bound Algorithm

CFRS: Cluster First, Route Second

DWD: Dantzig-Wolfe Decomposition

GA: Genetic Algorithm

GENI: Generalized Insertion Procedure

ISDA: Iterative School Decomposition Algorithm

LAR: Location-Allocation-Routing

RFCS: Route First, Cluster Second

SDA: School Decomposition Algorithm

SSDA: Simplified school decomposition algorithm

TS: Tabu search

US: Unstringing and Stringing

Others:

AvgTT: Average Travel Time Per Trip (minutes)

MaxTT: Maximum Travel Time Per Trip (minutes)

DD: Deadhead

LUB: Loose upper bound

SUB: strengthened upper bound

Chapter 1 INTRODUCTION

School bus routing and scheduling (SBRS) is of huge significance in students' transportation. The major task of SBRS is to transport students from their homes to schools and vice versa. For the public schools, it is usually run by the county's Department of Education and sometimes it is contracted to a third party vendor; while private schools run bus system usually on their own.

The transportation consists of morning (AM) trips and afternoon (PM) trips. These two trips routing plan are similar except that AM trips pick up students from their homes and drop them off at schools while PM trips pick up students at schools and drop them off at their homes. It is not only for the sake of efficiency but also equity that in real applications, one trip routing plan is solved and is replicated for the other. The student who is picked up first in the AM trips should also be the first one to get dropped off in the PM trips so that the overall ride time for all students in one trip is balanced.

From the operator's perspective, the objective is to provide school bus service with minimum cost while satisfying certain constraints. Such cost consists of two parts: bus purchase cost and operation cost. The operation cost includes drivers' salary, fuel cost, bus maintenance cost, etc. The major cost in this system is the bus purchase cost and drivers' salary cost while both costs are highly related to the number of buses used. The bus purchase cost is directly proportional to the number of buses. For drivers' salary cost, consider the fact that school buses only provide morning and afternoon service at certain times. A common approach is to assign one driver to each bus. Hence, finding the minimum number of buses is the main objective for the school bus routing and scheduling problem. The average annual cost for each school bus is from $50,000 to $100,000. With that high annual cost, finding the minimum number of buses is of highest priority to the operators.

1.1 Definition

School bus routing and scheduling problem (SBRS) is usually treated as two consecutive problems. First, school bus routing (SBR) problem and then school bus scheduling (SBS) problem. A few terms need to be clarified first.

- **A trip**: a sequence of stops that starts from a school and goes to several bus stops for the corresponding school (afternoon trip) and vice versa (morning trip) while satisfying capacity constraint and/or maximum ride time constraint if applied.
- **Routing plan**: school- individualized plan, which consists of a set of trips for the corresponding school that can transport all students from the school to their homes (afternoon trip) and vice versa (morning trip).
- **A block**: a sequence of compatible trips that can be served by one bus.
- **Blocking plan:** a plan that groups all trips into a minimum number of blocks while maintaining compatibility constraint.
- **Scheduling plan**: the assignment of the specific vehicles and service crew including drivers to accommodate the blocking plan.

Usually, the blocking plan, which specifies the sequence of compatible trips, is only a part of scheduling plan. Besides the blocking, scheduling plan also consists of the arrangement of the buses and drivers/crews to accommodate the blocking plan. Bus maintenance scheduling, drivers/crews working hours limit and shifting should all be considered in the scheduling problem. However, in the school bus problem, its unique daily operation property along with the fixed operation time in the morning and afternoon, significantly simplifies the problem. Assigning one bus and one driver to each block for daily operation is a good way to map the blocking plan to the scheduling plan. As a result, the blocking plan can provide the exact number of buses and drivers needed to handle the students' transportation service. Therefore, in this thesis, we would only consider blocking plan. We adopt the vague defination of 'scheduling problem' which includes both blocking and scheduling, but from a more accurate definition, the 'scheduling problem' in this thesis only refers to the blocking problem.

1.2 Trip Compatibility

Many papers blurred trips and blocks, but there exists a significant difference. A routing plan with a minimum number of trips does not always yield the scheduling plan with a minimum number of buses (blocks). So is the routing plan with minimum total travel time. A simple illustration is shown in Figure 1. In this example, there are six stops in total: S1, S2, and S3 are school bus stops for school 1 and S4, S5, and S6 are for school 2. School 1 dismisses at 2:00PM while school 2 dismisses at 2:30PM. The travel time (in minutes) for each pair is marked next to the arcs. Without considering the blocking, plan one is to minimize the number of trips (MinN). Assume that the number of students that are using school buses for school 2 exceeds the capacity of one bus; a minimum of two trips are required for school 2. Also, assume that one trip can serve all stops for school 1. Based on these assumptions, it is easy to see that the minimum number of trips is three. If plan two's objective is to minimize the total travel time (MinTT) for students, three trips are needed again. Minimum number of trips and minimum total travel time are both obtained with the following plan: trip 1 is School 1→S3→S2→S1 (40 minutes), trip 4 is School 2→S5→S4 (40 minutes), trip 5 is School2→S6→S7 (40 minutes) and the total travel time is 120 minutes. Trip 1 reaches school 2 is at 2:45 PM, which misses the dismissal time of school 2 (2:30PM). Hence, trip 1 cannot be compatible with any trip for school 2 (assume all trips for school 2 have to depart at 2:30PM). Three buses are needed – one bus for each trip.

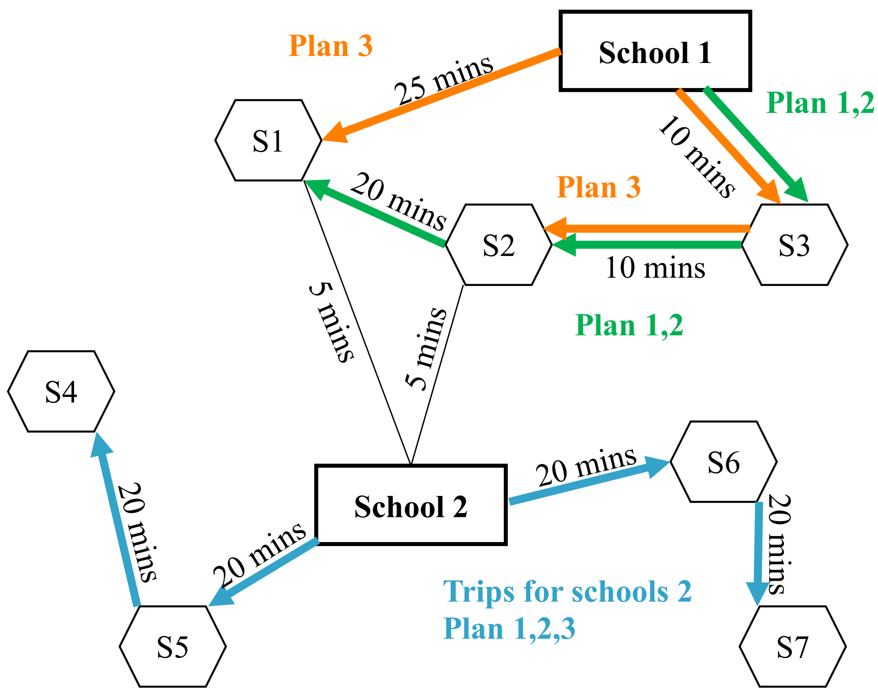

Figure 1 Example of solving routing and scheduling as related problems

Table 1 Summary of three plans for Figure 1

	Plan 1	Plan 2	Plan 3
Objective	MinN	MinTT	MinB
Number of trips	3	3	4
Total travel time	120 mins	120 mins	125 mins
Number of buses	3	3	2

Alternatively, consider the setting (Plan three aims at minimizing number of buses, MinB) in which trip 1 is replaced with trips 2 and 3 where trip 2 is school 1→S1 (25 minutes) → (school 2) and trip 3 is school 1→S3→S2 (20 minutes) → (school 2).

Trips 4 and 5 do not change. In this case, trips 2 and 3 both reach school 2 before 2:30 PM and therefore, can be compatible with the trips for school 2. The total buses needed is just two – each bus serves two trips (Bus 1: trip 2 → trip 4; Bus 2: trip 3 → trip 5).

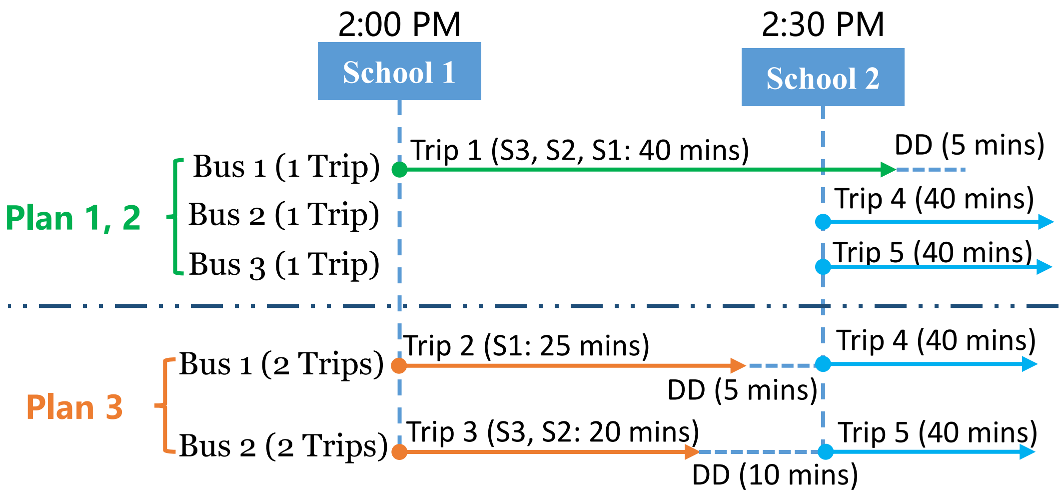

Figure 2 Graphical representation of scheduling of the example in Figure 1 (DD: deadhead)

The total travel time with students on board is 125 minutes, which is a slightly higher than plan one and plan two. The summary of three plans is listed in *Table 1*. The graphical representation of the scheduling of three plans is shown in Figure *2*. Notice that in Figure *2*, the deadhead (DD) is the sum of the travel time from the last stop of the previous trip to the initial stop for the next trip and the wait time at the that initial stop if the bus arrives earlier than the departure time. For instance, the deadhead of bus 2 in plan 3 is ten minutes (five minutes to drive from S2 to school 2 and 5 minutes to wait at school 2). Although minimizing the number of trips (Plan 1) and minimizing total travel time (Plan 2) are the most common objectives in use, they do not find the best solution with the minimum number of buses, which is more important from the provider’s perspective.

This example sheds light on a point that was significant but widely neglected – trip compatibility. It can be defined as follows:

- **Trip compatibility**: two trips are compatible if a bus has enough time to reach the initial location for the second trip after serving the first trip.

Trip compatibility is the interpretation of the result in this example. Plan 3 has two pairs of compatible trips (trip 2 is compatible with trip 4 and trip 3 is compatible with trip 5) while plan 1 and plan 2 have nothing. Therefore, even when plan 3 has more trips, it uses fewer buses. A relationship can be derived from this example is that the number of buses is the number of trips subtracted by the number of compatible trip pair.

1.3 Thesis Structure

In this chapter, the problem of school bus routing and scheduling is defined. The task, scope, objective, importance and some key terminologies of the problem are introduced (in Chapter 1.1). An example is presented (in Chapter 1.2) to show the importance of solving routing and scheduling problem simultaneously. Different representations of this example are well demonstrated. It shed light on the trip compatibility, which has been widely neglected from other researches.

In the next chapter, a literature review of school bus routing and scheduling is presented along with the research gap that is aimed to be filled by this thesis. A Mixed Integer Programming model is proposed in Chapter 3. An algorithm is developed in Chapter 4 to solve a relaxed version of the proposed model efficiently. In Chapter 5, computational tests are conducted to evaluate the model's performance along with the efficiency of the algorithm. At the end, the summary of this research is presented as well as the future research directions.

Chapter 2 LITERATURE REVIEW

School bus routing and scheduling (SBRS) problem has been a research topic for many years. The whole process of school bus system design, explained by Desrosiers et al. (1981), involves five steps: data preparation, bus stop selection, bus route generation, school bell time adjustment and route scheduling.

One approach is to combine the first two steps and solve it by location-allocation-routing (LAR) strategy or allocation-routing-location (ARL) strategy. The difference and application of these two strategies were explained in details by Park and Kim (2010). The major difference is that LAR will first assign students to stops and then generate the trips. But ARL will first group students into clusters considering capacity constraint, and then select bus stops. At the end, a trip is generated for each cluster. The third step is what widely referred to routing problem and the final step is the scheduling problem. Usually, the school bell time is pre-determined and works as a time window constraint rather than a decision variable. Most paper solve one or several parts of the whole school bus design problem consisting.

Due to the computational complexity, empirically school bus routing and scheduling problem (SBRS) is solved as two separate problems where the solution of the school bus routing problem (SBR) is the input for the school bus scheduling problem (SBS).

School bus routing problem (SBR) is a variation of Vehicle Routing Problem (VRP), which is a capacitated m-TSP (Traveling Salesman Problem with m trips) (Desrosiers et al., 1995). One difference between SBR and VRP is that in SBR, the travel time from the depot or the school to the first pick up stop is insignificant (Park and Kim, 2010) as well as the travel time from the last pickup stop to the school/depot. School bus providers mainly concern about the ride time for students rather than the total travel time for the buses. This makes SBR an Open Vehicle Routing Problem (OVRP). Moreover, it is a capacitated or distance (travel time) constrained OVRP (Bektaş and Elmastaş, 2007). The main difference between OVRP and general VRP is that the former problem tries to find a set of Hamiltonian paths rather than Hamiltonian cycles as for the classical VRP (Gendreau et al., 2008). However, as stated by Syslo, Deo and Kowalik (1983), the minimum Hamiltonian path problem is still an NP-hard problem because it can be transformed into a minimum Hamiltonian cycle problem, which is a well-known NP-hard problem.

The School Bus Scheduling (SBS) problem, as stated above, addresses the routing plan with a specific driver and vehicle at a specific time. It is usually separated from routing problem, but there exists one unique type of problem that combines these two problems, which is the routing problem with time window constraints. By incorporating time window constraints, the routing problem solves part of the scheduling problem. Although such a problem does not directly give us the blocking/scheduling plan, the scheduling problem becomes trivial (assignment problem) given the routing plan with a time window. Detailed information about school bus design papers are list in Table *2* including classification, objectives, constraints, data etc. Figure *3* shows the problem each paper solved and the papers listed in Table *2* but omitted in Figure *3* all solve school bus routing problem.

2.1 Classifications

In reality, public school bus service is usually run by county's department of education, which inherently implies the multi-school settings. However, quite a lot of papers focus on single-school problem due to its simplicity and similarity to the classic single-depot VRP. Even if dealing with the multi-school problem, many papers also decomposed it into the single-school problem by assuming each bus trip is exclusive for one school (Corberán et al., 2002). Still, some papers consider mix-load service for the multi-school problem. Braca et al. (1994) solved the SBR problem for multi-school bus routing in New York City by Location Based Heuristic method. It is an insertion-based algorithm which inserts the vertex with minimum insertion cost among all un-routed vertices into the current trip and repeats this procedure by starting at random vertex and chooses the best solution. Park, Tae, and Kim (2012) expanded Braca et al.'s research and solved the mixed load school bus routing and scheduling problem.

Another classification is urban school and rural area school bus routing and scheduling problem. In an urban area where there are generally more students in each stop, the bus capacity is usually the binding constraint (Braca et al., 1994) and stops may need to be served more than once. Therefore, maximum ride time constraint can be relaxed under certain conditions (Bowerman, Hall and Calamai, 1995). While in a rural area, where it is more common to have a few students at each stop, trips are relatively longer such that maximum ride time becomes the critical constraint. It also makes vans or smaller vehicles more economical than buses thanks to the large geographical area and small student population. Thus, vehicle type selection or mix fleet is preferred (Ripplinger, 2005).

To estimate the level of service for the school bus, Bowerman, Hall, and Calamai (1995) applied 3E (Efficiency, Effectiveness, and Equity) criteria proposed by Savas (1978). One example of equity is afternoon school bus trip sequence. An afternoon trip should be a replicated sequence (except the schools) of the morning trip to balance the total ride time for each student. The balance of maximum load and/or maximum ride time among all trips is another equity concern. They were also incorporated into the objective by some research (Li and Fu, 2002; Bowerman, Hall and Calamai, 1995).

Homogeneous and heterogeneous fleet involves the bus capacity. However, it is more than that. The degree of crowding, the allowance of standing are all influential factors to the heterogeneity of the buses. National Association of State Directors of Pupil Transportation Services (1999) regulates that maximal three young students (lower than the third grade) and two elder students (higher than the third grade) can seat in a typical 39-inch school bus seat.

2.2 Objectives

The most common objectives are to 1) minimize the total number of trips, 2) minimize total travel distance or travel time, or to 3) minimize the combination of the first two (listed in Table *2*). Simchi-Levi and Bramel (1990) stated that the solution with minimum total travel distance must use a minimum number of trips in a large network if the distance falls into the general norm. The balance of the maximum load and maximum ride time between different trips was also considered in the objective (Bowerman, Hall and Calamai, 1995; Li and Fu, 2002; Lima et al., 2017). Bowerman, Hall, and Calamai (1995) further incorporated student walking distance in their model.

Corberán et al. (2002) and Pacheco and Martí (2006) both tried to minimize the number of trips while minimizing the maximum ride time for students. These two objectives are conflicting because fewer trips would require longer trip length (Corberán et al., 2002). Laporte, Nobert, and Arpin (1986) included the depot cost in the objective function as they considered whether to use a potential depot as a decision variable. Some objectives contain the artificial cost. One instance is explained by Laporte, Nobert, and Taillefer (1988) that when formulating the problem as modified assignment problem, the different artificial cost could result in different objectives. Another important component is time window violation penalty (Taillard et al., 1997), capacity or maximum ride time constraints violation penalty (Gendreau, Hertz and Laporte, 1994). Tabu search could also use the penalty. To expand search range, the penalty is applied to the vertices that have frequently been moved (Glover, 1989; Gendreau, Hertz and Laporte, 1994). The detailed objectives are listed in Table *2*.

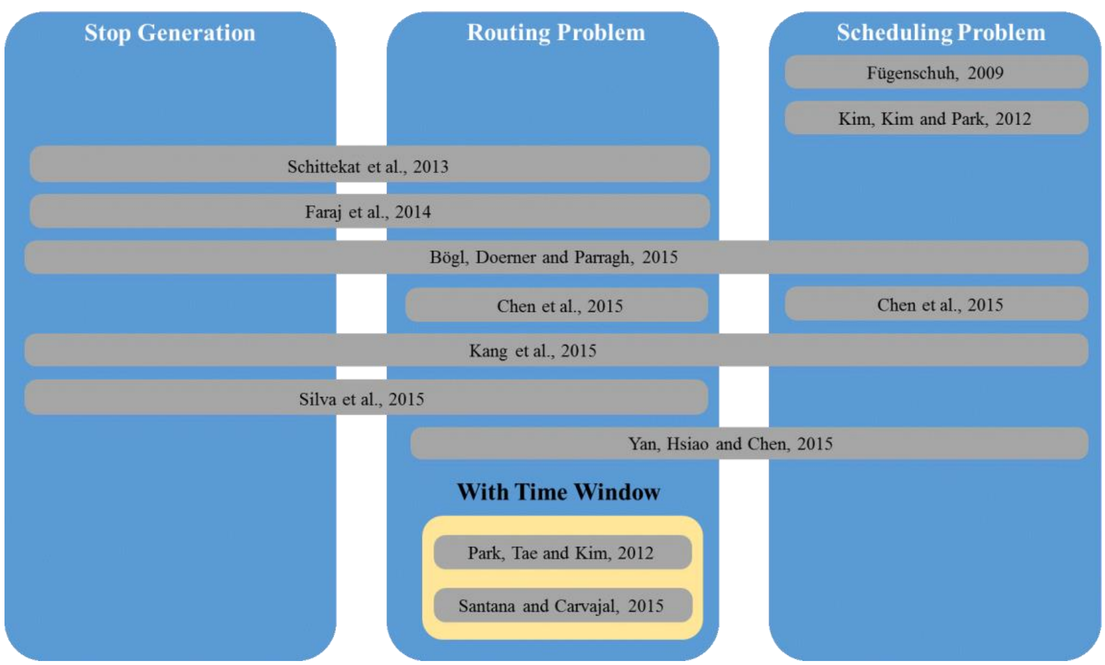

Figure 3 Problem coverage of school bus related papers

Table 2 Literature Review of SBRS

Author	Year	Objective	Constraints	# of School	Fleet Mix	Data
Newton and Thomas	1969	Not specified	LOG, C, MRT	S	HO	Artificial, up to80 stops
Angel et al.	1972	TTD, NOT	LOG, C, MRT	M	HO	1,500 students, Tippecanoe, Indiana
Bennett and Gazis	1972	TTD	LOG, C	S	HO	256 students, Toms River, New Jersey
Newton and Thomas	1974	TTD, NOT	LOG, C, MRT	M	HO	1,097 students, 76 stops
Verderber	1974	TTD, NOT	LOG, C, MRT	M	HO	11,000 students, New York
Gavish and Shlifer	1979	TTD, NOT	LOG, C, MRT	S	HO	Artificial, 21 stops
Bodin and Berman	1979	NOT	LOG, C, MRT, TW	M	HO	13,000 students, Brentwood, New York
Dulac, Ferland, and Forgues	1980	TTD, NOT	LOG, C, MRT, MWT	S	HO	585 students, 99 stops, Drummondville, Canada
Hargroves and Demetsky	1981	TTD, NOT	LOG, C, MRT, MNS	M	HT	8,537 students, Albemarle, Virginia
Chapleau, Ferland, and Rousseau	1985	NOB	LOG, TW	M	HO	37 schools, New Haven, Connecticut
Russell and Morrel	1986	TTD	LOG, C, MRT	M	HO	140 disabled student on nine bus routes
Desrosiers et al.	1986	TTD, NOT	LOG, C, MRT, MWT	M	HO	About 16,000 students, Drummondville, Canada
Chen et al.	1990	TTD, NOT	LOG, C, MRT, TW	M	HO	2,413 students, 40 buses, 48 routes, Choctaw, Alabama
Thangiah and Nygard	1992	TTD, NOT	LOG, C, MRT	S	HT	Nine buses, 315 students, 180 stops; Eight buses, 210 students, 116 stops
Braca et al.	1994	TTD, NOT	LOG, C, MRT, TW, MNS	M	HO	838 stops, 73 schools, Manhattan, New York
Bowerman, Hall, and Calamai	1995	TTD, NOT, SWD, BAL	LOG, C, MRT	S	HO	138 students, Ontario, Canada
Corberán et al	2002	NOT, MRT	LOG, C, MRT	S	HO	19 schools, from 2 to 14 buses, Burgos, Spain
Li and Fu	2002	TTD, NOT, BAL	LOG, C	S	HT	86 students, 54 stops, Hong Kong
Ripplinger	2005	TTD	LOG, C, MRT	S	HT	131 student, artificial
Spada, Bierlaire, and Liebling	2005	TTD	LOG, C,TW	M	HT	274 students, Switzerland
Schittekat, Sevaux, and Sörensen	2006	TTD	LOG, C, SWD, SBL	S	HO	Artificial, 50 students, ten stops
Pacheco and Martí	2006	NOT, MRT	LOG, C	S	HO	Benchmark problem from Corberán et al., 2002

Author	Year	Objective	Constraints	# of School	Fleet Mix	Data
Bektaş and Elmastaş	2007	TTD, NOT	LOG, C, MRT, TW, SBL	S	HO	519 students, central Ankara, Turkey
Fügenschuh	2009	NOB	SCH	M	HO	5 counties data from German, up to 490 trips, 102 schools
Díaz-Parra et al.	2012	TTD, NOT	LOG, C, MRT, SBL	S	HO	Artificial, 50 problems, each problem has 200 bus stop
Kim, Kim and Park	2012	NOB	LOG, SCH	M	HO/HT	Artificial, up to 100 schools, 562 trips and 28175 students
Park, Tae and Kim	2012	NOT	LOG, C, MRT, TW	M	HT	Artificial, up to 100 schools, 2000 stops, 32048 students
Schittekat et al.	2013	TTD	LOG, C, SBL	S	HO	Artificial, up to 80 stops, 400 students
Caceres, Batta, and He	2014	TTD, NOT	LOG, C, MRT,SBL	M	HO	Williamsville Central School District, 13 schools, up to 177 stops and 1237 student per school
Faraj et al.	2014	TTD	LOG, C, MRT	M	HT	944 students, 23 schools, Brazilian city
Kinable, Spidksma and Vandem Berghe	2014	TTD	LOG, C, MT	S	HO	Artificial, up to 40 stops and 800 students
Bögl, Doerner and Parragh	2015	TTD, PLT	LOG,C,SBL, MWD, TSF	M	HO	Artificial, up to 8 schools, 500 students
Chen et al.	2015	TTD, NOB	LOG, SCH	S	HO /HT	Benchmark problems from Park, Tae and Kim 2012; Kim, Kim and Park, 2012
Kang et al.	2015	TTD	LOG, C, MRT	M	HT	26 students, 6 schools, 3 buses
Kumar and Jain	2015	TTD	LOG, LOG	S	HO	Artificial, up to 40 schools, 235 trips, 11600 students
Mushi, Mujuni and Ngonyani	2015	TTD	LOG, C	S	HO	58 stops, 456 students, Dar es Salaam, Africa
Santana, Ramiro, and Romero Carvajal	2015	TTD	LOG, C, TW	S	HO	600 students, 440 nodes, Bogota, Colombia
Silva et al.	2015	TTD	LOG, C, SBL	M	HT	716 students, 23 schools, Brazilian city
Yan, Hsiao and Chen	2015	TTD, PLT	LOG, LOG	S	HO	Six universities (treated as one school), 400 students, Taiwan
Yao et al.	2016	TTD	LOG, C, MRT	M	HO	Artificial, up to 2 schools, 116 stops and 1088 students
de Souza Lima et al.	2017	TTD, NOT, BAL	LOG, C	M	HT	Artificial, up to 20 schools, 150 stops; Benchmark from Park, Tae and Kim 2012

Note: NOT: Number of trips; NOB: Number of buses; TTD: Total travel distance; MRT: Maximum ride time; PLT: Penalty; BAL: Balance between each trip; SWD: Student waiting time; LOG: Logistic constraints; C: Capacity Constraint; TW: Time window constraint; MWD: Maximum walk distance; SBL: Sub-tour elimination constraint; TSF: Transfer; SCH: Scheduling; S: Single-school; M: Multi-school; MNS: Minimal number of student to create a bus trip; HO: Homogeneous fleet; HT: Heterogeneous fleet

Table 3 Literature review of VRP

Author	Year	# of Depot	Fleet Mix	Objective	Constraints	Algorithm
Tillman	1969	M	HT	TTD	LOG	Clarke and Wright Saving
Held and Karp	1969	S	HO	TTD	LOG	Spanning tree
Wren and Holliday	1972	M	--	TTD	LOG, C, MRT	Saving, Sweep
Gillett and Miller	1974	S	HO	TTD	LOG, C, MRT	Sweep, 2 Phase
Lenstra and Kan	1975	S	HO	TTD	LOG, C, MRT, SBL	Assignment, B&B
Gillett and Johnson	1976	M	HO	TTD	LOG, C, MRT	Sweep
Christofides, Mingozzi, and Toth	1981a	S	HO	TST	LOG, C, MTT, SBL	Tree Search, K-means Clustering
Fisher and Jaikumar	1981	S	HO	TTD	LOG, C, SBL	Assignment, B&B
Laporte, Nobert, and Arpin	1986	M	HO	TTD, DC	LOG, SBL, CB	Integer Programming, B&B
Laporte, Nobert, and Taillefer	1988	M	HO	TTD	LOG, SBL, CB	Assignment, B&B
Desrosiers, Sauvé, and Soumis	1988	S	HO	NOB, TTD	LOG, C, TW	Lagrangian Relaxation
Desrochers, Desrosiers, and Solomon	1992	S	HO	TTD	LOG, C, TW	Column Generation, Partition
Osman	1993	S	HO	TTD	LOG, C, MRT	Saving, Simulated Anneal, TS
Gendreau, Hertz, and Laporte	1994	S	HO	TTD	LOG, C, MRT	TS, Insertion, 2 Phase
Renaud, Laporte, and Boctor	1996	M	HO	TTD	LOG, C, SBL	TS
Taillard et al	1997	S	HO	TTD, ELP	LOG, TW	TS, 2/3-opt
Prins	2004	S	HO	TTD	LOG, C, MRT	TS
Díaz-Parra et al	2012	S	HO	TTD, NOB	LOG, C, MRT, SBL	Genetic Algorithm, K-means Clustering

Note:

S: Single-depot
M: Multi-depot
HO: Homogeneous Fleet
HT: Heterogeneous Fleet
TTD: Total Travel Distance
NOB: Number of Buses
DC: Depot Cost
TST: Total Service Time + node service time
ELP: Early and Late Penalty
LOG: Logistic Constraint
C: Capacity Constraint
MRT: Maximum Ride Time
SBL: Sub-tour Elimination Constraint
CB: Chain Barring Constraint
TW: Time Window Constraint
TS: Tabu Search
B&B: Branch and Bound Algorithm

2.3 Constraints

2.3.1 Logistic Constraints

Logistic (or degree) constraints are the basic VRP constraints that define trips. They regulate the conservation of flow: 1) a specific vehicle entering a node must leave this node; 2) each trip starts from a school (or depot) and goes back to the school (or depot); 3) all vertices have to be served (either single-visit or multi-visit). Although SBR is an OVRP, it is easier to form the round trips starting from and going back to the school and then exclude the first (for morning trips) or the last (for afternoon trips) edge. For a single-visit, single-school problem, logistic constraints are the assignment problem constraints.

2.3.2 Capacity Constraints

Capacity constraint is one of the most important constraints not only for SBRS but also for more general Vehicle Routing and Scheduling Problem. It distinguishes the VRP from m-TSP. Therefore, a capacity relaxation of VRP will yield an m-TSP. Such relaxation might be valid under the condition where capacity is not the binding constraint, like the rural area school bus problem as mentioned above. Other constraints can also be integrated within capacity constraint framework, like sub-tour elimination constraints, which will be discussed in Chapter 2.3.4.

The fundamental capacity constraint is the maximum number of passengers (student for the SBRS) on each bus. The difference of the capacity directly determines the homogeneity or the heterogeneity of the fleet size. However, there is no Federal regulation about the specific maximum number of student per bus. For a standard school bus, the maximum capacity is 72 students. But the safety concern often drives the authorities to lower such capacity. The general guideline is that three younger students or two elder students can fit in one standard 39-inch seat. This allowance of crowding significantly influences the bus capacity. Another part of capacity constraints occurs at a multi-visits setting where each stop can be visited more than once, and only some of the students at one stops are picked up for one trip. Under such circumstances, capacity constraints also include the decisions of 1) how many students should be picked up at one stop for one trip, 2) how many times each stop needs to be visited, which implies that all students at all stops have to be picked up.

2.3.3 Time Window Constraints

Time window constraints can be either incorporated in routing problem or scheduling problem. An important classification of time window constraints is the hard time window or soft time window. The hard time window is that any violation of time window is prohibited while soft time window is that violation of time window is allowed at the expense of a penalty. Common time window constraints include the following three parts: 1) a trip has to depart from and goes back to the depot (schools) at a certain time window; 2) maximum travel time for each trip; 3) a vertex has to be served within a specific time window. By incorporating these time window constraints into routing problem, each trip is assigned with a specific start time and end time. The blocking and scheduling problem given such routing plan with time windows becomes a simple assignment problem. In the field of research, Braca et al. (1994) limited the school buses arrive at schools no earlier than 25 minutes before and no later than 5 minutes after the school bell times.

Maximum ride time constraint is another common constraint in school bus problem. It is especially important for rural area SBRS as explained earlier. Some routing papers include the maximum ride time constraints although they do not consider all the time window constraints. Russell and Morrel (1986) used 45 minutes as the maximum ride time, Chen et al. (1990) limited the ride time to be less than or equal to 75 minutes. Park, Tae, and Kim (2012) tested 45 minutes and 75 minutes as the maximum ride time for several benchmark problems. The maximum ride time constraint always comes along with triangle inequality with respect to travel time, or distance of each arc (Solomon, 1987). It guarantees that when the maximum ride time constraint is violated for a trip, it is impossible to insert more vertices into that trip (Prins, 2004), which is the backbone for insertion algorithm that will be discussed in Chapter 2.4.2 and 2.5.2.

2.3.4 Sub-tour Elimination Constraints

Sub-tour elimination constraint is an essential constraint that prevents the formation of the illegal trips that are not connected to the school or the depot (Bektaş and Elmastaş, 2007). One of the most common sub-tour elimination constraints is borrowed from TSP. The idea is that for any subset (say S) of the graph (except trivial vertices), there should exist at least one edge connecting some vertices in that subset (S) to vertices outside of the subset. For the school bus routing problem, which falls into VRP category (which has the capacity constraint that TSP does not). The capacity constraint makes the formulation a little bit different: an appropriate lower bound is set on the number of trips required to visit all vertices of any subset of the graph (except trivial vertices) in the optimal solution. The appropriate lower bound is calculated based on the capacity constraint (Laporte, 1992).

Another common approach is to formulate sub-tour elimination constraints as the artificial commodity flow. It treats the routing problem as a water flow. The flow variable at each vertex, or the degree of the flow, increases by one at vertices that are visited by a trip (Bowerman, Hall, and Calamai, 1995). The flow variable value is strictly one greater than the leading vertex. The only exception is the school (depot) so that if a trip does not connect to the school, it will violate the constraints. For instance, one sub-tour is: stop 1 → (0) → stop 2 (1) → stop 3 (2) → stop 1 (3), where the numbers in parenthesis are flow variable. Since stop one cannot be degree 0 and degree 3 simultaneously, such illegal trips are identified.

Another variation of the conservation of flow is proposed by Miller, Tucker, and Zemlin (1960). The benefit of their formulation is that it can be incorporated into capacity or travel time constraints framework such that the logistic, capacity/maximum ride time and sub-tour elimination constraints can be replaced by two set of constraints. The formulation can be seen from Desrosiers, Sauve, and Sourmis (1988) and Bektaş and Elmastaş (2007).

2.3.5 Chain-barring Constraints

A Chain -barring constraint only occurs in multi-school (multi-depot) scenarios, in which it aims to eliminate the illegal trips that start from one school (depot) and end at another school (depot). The detailed constraints formulation is given in Laporte, Nobert, and Arpin (1986). However, in SBRS, stops are usually assigned to only one school. It is impossible to form trips starting from and ending at different schools if trips are limited within each school. Empirically, this constraint is redundant for SBRS. For the sake of saving running time, chain barring constraint is excluded in our formulation.

2.3.6 Scheduling Constraints

Two common unique constraints for blocking and scheduling problems are trip compatibility constraint and drivers' maximum legal working hours constraints. The trip compatibility constraint, as explained above, is that whether a bus has enough time (deadhead) to reach the initial location of the second trip after serving the first trip. The drivers' working hours constraints are that drivers cannot work continuously longer than the maximum legal working hours without a certain amount of break. There are two approaches to handling this drivers' working hours constraint under the routing problem framework. The first is to simply increase the travel time for each trip so that the additional travel time can be used as the break for the driver (Brandao and Mercer, 1997). The second approach is to treat the break as dummy vertices with certain serving duration (Rochat and Semet, 1994). The drivers' break time, no matter from additional travel time or dummy break vertices can be broken into several pieces, like the deadhead between trips. But as explained before, school buses' daily operation property guarantee such constraint hold under daily operation routine.

2.4 Formulations and Algorithms for SBR

Both exact and heuristic algorithms are proposed to solve school bus routing problem (SBR). Since the algorithms are highly related to the formulation of the model, we discuss the model and algorithm together.

2.4.1 Exact Algorithms

1) Assignment Problem

One way to formulate SBR is as a modified assignment problem. Lenstra and Kan (1975) proposed a transformation method, which transforms m-TSP (Traveling Salesman Problem with m trips) into a 1-TSP. Christofides, Mingozzi, and Toth (1979) then proved that m-TSP with the capacity constraint is the solution for VRP with m trips. Based on that, Laporte, Mercure, and Nobert (1986) proposed the modified assignment problem formulation using the m-TSP to 1-TSP transformation method. That method involves three steps: 1) choose an upper bound on m (number of trips); 2) introduce artificial depots such that each route is “assigned” to one depot; 3) extend the vertices set and its corresponding distance matrix. Different cost extension strategies for artificial edges can result in different objectives. By doing these, SBR can be formulated as a modified assignment problem with relaxing sub-tour elimination constraint. More detailed explanation can be found in Laporte (1992). Bektaş and Elmastaş (2007) also applied this dummy depot idea in SBR and transferred it into a problem of finding m node-disjoint paths between two points. Branch and Bound (B&B) algorithm is the classical way to solve assignment problem while Desrosiers, Sauve, and Sourmis (1988) used Lagrangian relaxation to find an appropriate lower bound for the problem.

2) Set Partitioning

Another unique formulation is proposed by Balinski and Quandt (1964), which is called the set partitioning formulation. It was adopted by Desrosiers, Soumis, and Desrochers (1984), Desrochers, Desrosiers, and Solomon (1992) and Xu et al. (2003). The decision variable is that whether a trip among all feasible trips is used in the optimal solution and that whether a stop appears on the trip. It works for the small size problems, but the number of columns would increase exponentially with the problem size so that the model cannot be solved directly (Laporte, 1992). Desrochers, Desrosiers, and Solomon (1992) used column generation and dynamic programming to solve this model while Xu et al. (2003) proposed a two-step algorithm combining column generation and a heuristic method to solve it.

3) Other Exact Algorithms

Dynamic programming has been applied to VRP since the 1970s (Desrosiers, Dumas and Soumis, 1986; Desrosiers and Soumis, 1988). State-space relaxation proposed by Christofides, Mingozzi, and Toth (1981b) is an efficient way to reduce the number of states. The most recent application of dynamic programming is conducted by Mahmoudi and Zhou (2016) to solve a pickup and delivery problem with time windows. Tree search algorithm is also applied to VRP (Held and Karp, 1969; Christofides, Mingozzi, and Toth, 1981a), which finds the k-degree spanning tree where k is the number of edges that are incidents to the depot. But Desrosiers et al. (1995) claimed that Dantzig-Wolfe decomposition/column generation and Lagrangian relaxation outperform K-tree method even at scenarios when k-degree spanning tree should perform at its best.

2.4.2 Heuristic Algorithms for SBR

1) RFCS and CFRS

One classification for heuristics for SBR are Route First, Cluster Second (RFCS) (first proposed by Beasley,1983) or Cluster First, Route Second (CFRS) (Bodin and Berman, 1979). Route First, Cluster Second Algorithm is to develop one long trip as 1-TSP after relaxing capacity and maximum ride time constraints. And then find the optimal partition that cuts this one long trip into several smaller trips which satisfy the capacity and maximum ride time constraints. One example of RFCS method is Space Filling Curve with Optimal Partitioning, proposed by Bowerman, Hall, and Calamai (1995).

Cluster First, Route Second (CFRS) algorithm first groups stops into different clusters while making sure the capacity constraint holds for each cluster. The second step is to solve minimum weight TSP for each cluster. If the maximum ride time constraint is violated, the algorithm tries to re-insert some stops into its neighboring clusters to find a feasible solution. Some improvement mechanisms are required to change boundary vertices to the neighboring cluster and solve the second step iteratively to find the optimal solution.

2) Trip Generation

Trip generation is the backbone for many heuristic algorithms like Insertion Method, Tabu Search, etc. The most common classification of trip generation procedure is the sequential procedure and the parallel procedure. Sequential procedure builds one trip at a time while parallel procedure builds multiple trips together. One example of the sequential procedure is Location Based Heuristic proposed by Bramel and Simchi-levi (1992). Christofides, Mingozzi, and Toth (1979) developed a two-phase algorithm that combines these two route generation procedures. In phase one, sequential route generation is adopted which inserts an un-routed vertex i with minimum insertion cost to the current generating trip and then uses r-opt algorithm (which will be defined in Chapter 2.4.2, 3) Tabu Search) to optimize this trip. Phase two is the parallel route construction, which inserts an un-routed vertex to its minimum insertion cost trip until all vertices are assigned, then optimize each trip by r-opt.

A Generalized Insertion Procedure (GENI) proposed by Gendreau, Hertz, and Laporte (1992) is a widely adopted route generation algorithm. It considers two types of insertions with two orientations, which are equivalent to 3-opt and 4-opt exchange insertion mechanisms. The post-improvement algorithm, Unstringing, and Stringing (US) were also proposed along with GENI.

3) Tabu Search

Tabu search (TS) is the one of most adopted heuristic algorithm to solve SBR. Prins (2004) claimed that Tabu Search (TS) is the best metaheuristics for VRP which easily outperforms Simulated Annealing, Genetic Algorithm and Ant Algorithms (Gendreau, Laporte and Potvin, 1998; Golden et al., 1998 and Osman, 1993). Tabu Search consists of three parts: 1) find an initial feasible solution; 2) iteratively improve the solution which involves strategies to jump out of local optima and expand search area; and 3) stop criterion. The Clarke and Wright Saving Method is the most widely used algorithm to form an initial solution, not only for TS but also for some other heuristic algorithms. The procedure starts with n trips, each trip starts from the school, serves one stop and goes back to the school. The algorithm then computes the saving for serving vertices i and j with one trip (the trip sequence is denoted as school → i → j → school) for all vertices in the graph. The highest saving pair is merged and then the second, so on and so forth, until capacity or maximum ride time constraint is violated.

The most common trip improvement mechanisms are edge exchanges and chain exchanges (Babin, Deneault and Laporte, 2005). R-opt is the most famous edge exchange mechanism, which removes r edges from the current trip and finds a better reconnection in the remaining trips. When r equals to 2, it becomes the 2-opt which is especially effective to delete cross in a trip (Lin, 1965). The best-known chain exchange mechanism is called Or-opt (Or, 1976), which moves three consecutive vertices (at a replicated or reversed sequence) to a different location, then two consecutive vertices and finally one vertex exchange. The detailed information and improvement of trip exchange are described in Babin, Deneault, and Laporte (2005). There are two widely used selection strategies: best-improvement strategy and first-improvement strategy (Osman, 1993). The former is to select the best solution in the neighborhood while the latter is to accept the first solution that satisfies the acceptance criterion in each improvement iteration. For stop criterion, TS usually stops when a maximum number of iterations is reached after the best solution has been found. Osman (1993) proposed to calculate the best stopping number of iteration which saves extra running time and is still able to find the best solution.

4) Sweep Algorithm

Sweep Algorithm is a sequential trip generation procedure and was first proposed by Wren and Holliday (1972). It was quite popular for VRP in 1970s-1980s. It works like this: first, it represents the graph in a polar coordinate and then uses a radial arm to sweep the whole graph, inserts the swept vertex in the current generating trip until its capacity or maximum ride time constraint violated. Each trip is then solved by 1-TSP, and the boundary vertices are exchanged to the neighboring trips to see if a better solution exists. Gillett and Miller (1974) further developed this method by repeating this trip generation process from different start vertex and in both clockwise and counter-clockwise direction.

5) Genetic Algorithm

Thangiah and Nygard (1992) adopted Genetic Algorithm (GA) to solve SBR. It used the genetic sectoring method to group stops into several clusters in a polar coordinate system, serving as the first step in Cluster First, Route Second method. Prins

(2004) proposed another hybrid GA to solve VRP, which used a sequence of vertices and trip delimiter as chromosomes. It is the second phase of Route first, Cluster Second method. One advantage of this algorithm is the flexibility. It works for different scenarios: minimizing total cost or minimizing the number of trips, mixed (heterogeneous) vehicle fleet problem, or even with a limited number of vehicles. Kang et al. (2015) adopted GA to solve a multi-school heterogeneous problem. Other application includes Diaz-Parra et al. (2012). However, GA is worse than many Tabu Search algorithms with respect to finding the optimal solution (Prins, 2004).

2.5 Formulations and Algorithms for SBRS

2.5.1 Exact Algorithms

Various formulation for SBRS has been well summarized in Desrosiers et al. (1995). On the one hand, the formulation of SBRS is not a big task as it can be considered as the combination of routing and scheduling problem. On the other hand, the convex hull the problem is still not well formulated. As a result, no formulation is significantly stronger than others. The problem becomes how to solve the SBRS problem.

Two of the most common approaches to modify the model to solve the problem optimally is by 1) Dantzig-Wolfe decomposition/column generation (DWD) (Desrosiers, Soumis and Desrochers, 1984; Kinable, Spidksma and Berghe, 2014; Santana and Carvajal, 2015) and 2) Lagrangian relaxation (Desrosiers, Sauvé and Soumis, 1988; Lamatsch, 1992; Mesquita and Paixão, 1992). Dantzig-Wolfe decomposition is better than Lagrangian relaxation regarding the solution quality and running time. The reasons explained by Desrosiers et al. (1995) are three. First, the simplex algorithm to solve the master problem for Dantzig-Wolfe decomposition uses more information and thus is more efficient. Second, DWD allows the use of many heuristics to converge rapidly. Third, DWD provides more information to construct better branch-and-bound problem to explore the internality gap.

After model modification (Dantzig-Wolfe decomposition/column generation or Lagrangian relaxation), branch-and-bound (B&B) is applied to the problem. An appropriate lower bound is usually the key to efficiently improve the performance of B&B. Kontoravdis and Bard (1995) proposed three lower bounds for routing and scheduling problem. The first one is derived from the capacity constraint. The second lower bound is the maximum clique of incompatibility graph. Such incompatibility can be caused by either capacity or time window constraint. The third lower bound is derived from the time window constraint.

2.5.2 Heuristic Algorithms for SBRS

Due to the complexity of the model and limitation of the exact algorithms, more recent work concentrated on trip generating algorithms, which can efficiently generate feasible routing plan and are easy for improvement. The earliest and one of the most important attempts to simultaneously solve routing and scheduling problem by trip generating algorithm is conducted by Solomon (1987).

1) Time-oriented Insertion Algorithm

Solomon (1987) expanded several routing heuristic algorithms (saving algorithm, insertion algorithm, nearest neighbor algorithm, and the sweep algorithm) to solve school bus routing and scheduling simultaneously. The advanced time-oriented saving method is an expansion of traditional saving method by incorporating the urgency of customer or the ‘time closeness’ of a customer to another into the saving criteria. The advanced time-oriented sweep algorithm treated the traditional sweep method (explained in Chapter 2.4.2) as the first clustering step and then solved the scheduling problem for each cluster with some unscheduled vertices due to the time window constraint. Among these heuristics, Solomon claimed that the two-stages time-oriented insertion method outperformed the others.

The major task of insertion method is to find the “best” insertion point in the “best” generating trip for the un-routed vetices. The most important aspect is how to select the “best” insertion point. One of the common approaches is selecting the “best” which has the highest evaluation criterion. Such criterion could be the minimum additional distance and time or the maximum saving derived (Solomon, 1987). Another approach is to select the “best,” of which has the largest absolute gap between the lowest and second lowest insertion cost point. Such “regret” mechanisms chose the one that would have much larger insertion cost if not immediately inserted. The feasibility check of inserting a vertex is also of huge importance since such insertion may alter the service time for all following vertices. Necessary and sufficient conditions for time feasibility check is proposed by Solomon (1987), which is more efficient to check the time feasibility than explicitly checking all vertices.

Potvin and Rousseau (1991) expanded Solomon’s algorithm and proposed to use a parallel trip generating algorithm to solve vehicle routing and scheduling problem. One of the benefits of this algorithm is that it can greatly improve the poor quality of the last trip, which is quite common in a sequential insertion method.

2) Post-improvement Mechanisms

Post-improvement is the final step of trip generation heuristics including insertion method. One of the recent application is from Park, Tae, and Kim (2012) where the post-improvement tries to insert stops to another trip with lower cost and maintain feasibility. Most of the improvement algorithms are the extensions or the variances of 2-opt, 3-opt (Lin, 1965) and Or-opt procedure (Or, 1976) which has been discussed in Section 2.4.2. It usually examines whether removing and reconnecting one, two or three adjacent vertices into a different location on the same trip or another trip would yield a better result. The most important concept of the post-improvement mechanism is the examination scope. It will increase the chance to jump out of local optima and find the global optima if we are allowed to move more vertices at one time, allowing between-trip improvement and increasing the total maximum number of iterations (or the maximum number of iterations after the current best solution has been found). However, by contrast, more processing time is required. Considering that such examination is random and non-directional, it does not pass any information from one iteration to the next. The longer examination time does not always improve the solution, and it becomes inefficient. Thus, many papers incorporated the post-improvement into Tabu Search framework (Potvin, Kervahut and Rousseau, 1992; Renaud et al., 1996; Taillard et al., 1997; Badeau et al., 1997; Ngonyani, Mujuni and Mushi, 2015) where the revised move is forbidden for a number of iterations (see the detailed discussion in Section 2.4.2).

3) Other Heuristics

Another interesting method is proposed by Kontoravdis and Bard (1995), which is a greedy randomized adaptive search procedure. The greedy randomized means selecting vertices according to their potential saving but with some randomness to generate trips. Brandao and Mercer (1997) solve the VRS in a three-phase algorithm, which can take advantage of Tabu search and GENI (Gendreau, Hertz, and Laporte, 1992). Park, Tae, and Kim (2012) formulated the mixed-load SBR as a pickup and drop-off problem. The initial single load problem was solved by a modified sweep algorithm. Then the scheduling problem was solved as an assignment problem. At the end, a post-improvement algorithm was adopted to insert stops from one route to another, making the load to be mixed. Another unique work is from Caceres, Batta, and He (2014) where they incorporated the stochastic demand and uncertainty of the travel time into the school bus routing with time window constraint.

2.6 Research Gap

Also mentioned in Chapter 1, most research treated the school bus routing and scheduling as two separate problems. However, such separation will lead to poor-quality solution using more buses. A few efforts have been put to solve school bus routing and scheduling problem simultaneously (Bögl, Doerner and Parragh, 2015; Kang et al., 2015; Yan, Hsiao and Chen, 2015). But there still exist some research gap that has not be thoroughly studied by previous researches:

1) The mathematical formulation for school bus routing and scheduling problem is not well defined;

2) An efficient algorithm is missed to solve school bus routing and scheduling problem. The current algorithm can only solve "simple" problem ("simple" problem is referred to either single school problem or small size problem with a small number of stops, schools and students)

As a result, the contribution of this thesis is to formulate multi-school bus routing and scheduling problem, develop an efficient algorithm to solve the problem and evaluate the performance of the proposed model and algorithm.

2.7 Chapter Conclusion

In this chapter, an abundant review of school bus routing and scheduling problem (SBRS) is presented. It summarized the existing research of SBRS along with similar vehicle routing and scheduling problem from 1969 to now. It started (in Chapter 2.1) with different classifications of SBRS and their unique properties. Then the SBRS is analyzed by its objectives (in Chapter 2.2), constraints (in Chapter 2.3) and formulations and algorithms (in Chapter 2.4 and 2.5). The research gap is discussed (in Chapter 2.6), which is also the main contribution of this thesis.

Chapter 3 MODEL FORMULATION

A mixed integer programming model is developed to solve the school bus routing and scheduling problem. Due to the complexity of solving the original problem, a relaxed version of the original model is introduced. An efficient heuritisc algorithm, called School Decomposition Algorithm (SDA), is proposed in Chapter 4 to solve relaxed version of the original model. The basic assumptions of this model are:

- One organization runs the multi-school bus system, and its interest is to provide service to satisfy all students transportation need with minimum cost.
- All buses have the same capacity (homogeneous fleet).
- Each stop is assigned to one school so that mixed load is prohibited.
- Each school has a set of stops assigned to it. The assignment, number of students at stops and school start/dismissal time are all pre-known (bus stop generation is not considered in this model).
- Each stop has a number of students to be served. All stops and all students must be served. Stops can be visited more than once (a multi-visit is allowed).
- Trips are limited to schools while buses can run between schools. Each bus must finish one trip before heading to the next trip.

This problem is assumed to be a problem for a public school system, from which the first multi-school assumption comes. The county's Department of Education is assumed to be the organization that runs this system. Homogeneous fleet assumption makes the problem much easier, especially when we consider trip compatibility. And, it is a realistic assumption as the school buses for public schools are standardized. Note that the fleet heterogeneity discussed above mostly comes from the differences in the allowance of crowdedness for young and elder students. But the school buses are the same. Many studies will only allow single-visit for stops as typical TSP. However, in reality, multi-visit is quite common for urban school systems where each stop has many students. The final assumption prohibits a case that a bus goes to the next school with some students from the previous school onboard. Overall, these assumptions are all realistic.

3.1 Notation

The summary of notation is listed in *Table 4*. Five sets of binary decision variables and five sets of continuous decision variables are used to formulate the model. It has been discussed previously that replicating the afternoon trip to the morning trip not only is efficient but also balances the total ride time for students. In this problem, we solved afternoon trip with the objective to minimize the number of buses used.

Table 4 Notation summary of variables and parameters

Variables for school bus routing	
Variable name	**Description**
$s2t_{s,t}$	Binary decision variable that equals 1 if stop s is assigned to trip t
$t2s_{t,k}$	Binary decision variable that equals 1 if trip t is assigned to school k
$x^t_{s1,s2}$	Binary variable that equals 1 if in trip t the bus goes directly from stop $s1$ to stop $s2$
$b_{t1,t2}$	Binary variable that equals 1 if trips $t2$ can be served after trip $t1$ (they are compatible)
$l_{s,t}$	Binary variable that equals 1 if the last stop of trip t is stop s
$p4t_{s,t}$	Portion of the capacity of the bus doing trip t that is filled at stop s
tt_t	Travel time (duration) of trip t
end_t	The end time of trip t
$dd_{t1,t2}$	The travel deadhead duration from the last stop of trip $t1$ to the first stop of trip $t2$
$c^t_{s1,s2}$	The units of "artificial commodity" that is shipped from stop $s1$ to $s2$ by trip t (use for sub-tour elimination constraints, see Chapter 2.3.4)

Parameters for School bus routing and bus blocking	
Parameter name	Description
$Schools$	Set of schools
$Trips\_School_k$	Set of possible trips dedicated to school k
$Trips$	Set of all trips
$Stops\_School_k$	The set of stops in which students for school k should go to / come from
$Stops$	Set of all stops
Cap	The capacity of each bus
$Students_{s,k}$	The number of students at stop s for school k
O_k	School location for school k
$D_{s1,s2}$	The duration to drive from stop $s1$ to $s2$ plus the dwell time required at the stops
$start_t$	The start time of trip t, which is the dismissal time for the corresponding school
M	A large positive value (big-M)
C_T	Coefficient for total travel time = 1 for the cases solved
C_C	Coefficient of trips = 1000 for the cases solved
C_B	Coefficient of compatibility = changing in different cases
A	The number of additional allowed trips (See Chapter 5)
MRT	Maximum ride time

3.2 Model Formulation

The proposed mixed integer programming model is presented as follows. The objective is to minimize the number of buses required while minimizing total travel time. Minimum number of buses is achieved by minimizing the number of trips while maximizing trip compatibility. The coefficient assigned to total travel time is extremely small compared to the number of trips and trip compatibility. It makes sure minimizing the number of buses has higher priority than minimizing total travel time. The desired solution is the one with minimum total travel given the minimum number of buses.

$$Min\ Z = C_C \sum_{k \in schools} \sum_{t \in Trips\_Schook_k} t2s_{s,k} - C_B \sum_{t1 \in Trips} \sum_{t2 \in Trips} b_{t1,t2} + C_T \sum_{t \in Trips} tt_t \quad (1)$$

The constraints listed below are in charge of building trips for each school. Each trip is built from stops.

$$s2t_{s,t} \leq t2s_{t,k} \;\; \forall k \in Schools, s \in Stops\_School_k, t \in Trips\_Schook_k \tag{2}$$

$$\sum_{s2 \in Stops\_School_k | s2 \neq s} x^t_{s,s2} = s2t_{s,t} \;\; \forall k \in Schools, s \in Stops\_School_k, t \in Trips\_Schook_k \tag{3}$$

$$\sum\nolimits_{s \in Stops\_School_k} x^t_{O_k,s2} = t2s_{t,k} \;\; \forall k \in Schools, t \in Trips\_Schook_k \tag{4}$$

$$\sum_{j \in Stops\_School_k} x^t_{s,j} = \sum_{i \in Stops\_School_k} x^t_{i,s} \;\; \forall k \in Schools, s \in Stops\_School_k, t \in Trips\_Schook_k \tag{5}$$

$$\sum\nolimits_{t \in Trips\_Schook_k} t2s_{t,k} \geq \left\lceil \sum\nolimits_{s \in Stops\_School_k} Students_{s,k} / Cap \right\rceil \;\; \forall k \in Schools \tag{6}$$

$$\sum\nolimits_{t \in Trips\_Schook_k} t2s_{t,k} \leq \left\lceil \sum\nolimits_{s \in Stops\_School_k} Students_{s,k} / Cap \right\rceil + \mathrm{A} \;\; \forall k \in Schools \tag{7}$$

$$t2s_{t,k} \leq \sum_{s \in Stops\_k | s \neq O_k} s2t_{s,t}, \forall k \in Schools, \forall t \in Trips\_Schook_k \tag{8}$$

$$\sum\nolimits_{s \in Stops\_School_k} p4t_{s,t} \leq 1 \;\; \forall k \in Schools, t \in Trips\_Schook_k \tag{9}$$

$$\sum\nolimits_{t \in Trips} p4t_{s,t} = Students_{s,k} / Cap \;\; \forall k \in Schools, s \in Stops\_School_k \tag{10}$$

$$p4t_{s,t} \leq s2t_{s,t} \forall k \in Schools, s \in Stops\_School_k, t \in Trips\_Schook_k \tag{11}$$

$$\begin{gathered} \sum\nolimits_{i \in Stops\_School_k | i \neq s} c^t_{i,s} - \sum\nolimits_{j \in Stops\_School_k | j \neq s} c^t_{s,j} = s2t_{s,t} \\ \forall k \in Schools, t \in Trips\_Schook_k, s \in Stops\_School_k \mid s \neq O_k \end{gathered} \tag{12}$$

$$c^t_{s1,s2} \leq M \times x^t_{s1,s2} \;\; \forall k \in Schools, t \in Trips\_Schook_k, s1, s2 \in Stops\_School_k \tag{13}$$

$$t2s_{t1,k} \geq t2s_{t2,k} \;\; \forall k \in Schools, t1, t2 \in Trips\_Schook_k \mid t1 < t2 \tag{14}$$

$$tt_t = \sum_{s1 \in Stops\_School_k} \sum_{s2 \in Stops\_School_k | s2 \neq O_k} x^t_{s1,s2} \bullet D_{s1,s2} \;\; \forall k \in Schools, t \in Trips\_Schook_k \tag{15}$$

$$x^t_{s,O_k} = l_{s,t} \;\; \forall k \in Schools, s \in Stops\_School_k, t \in Trips\_Schook_k \tag{16}$$

$$end_t = start_t + tt_t \;\; \forall t \in Trips \tag{17}$$

$$end_{t1} + dd_{t1,t2} - M \times \left(1 - b_{t1,t2}\right) \leq start_{t2} \;\; \forall t1, t2 \in Trips \tag{18}$$

$$dd_{t1,t2} \geq \frac{M}{2} \times \left(1 - t2s_{t1,k1}\right) + \frac{M}{2} \times \left(1 - t2s_{t2,k2}\right) + \sum_{s1 \in Stops\_School_k} D_{s1,O_{k2}} \times l_{s1,t1} \tag{19}$$

$$\forall k1, k2 \in Schools, s1 \in Stops\_School_{k1}, t1 \in Trips\_Schook_{k1}, t2 \in Trips\_Schook_{k2}$$

$$\sum_{j \in Trips | j \neq t} b_{t,j} \leq 1 \; \forall t \in Trips \tag{20}$$

$$\sum_{i \in Trips | i \neq t} b_{i,t} \leq 1 \; \forall t \in Trips \tag{21}$$

$$tt_t \leq MRT \; \forall t \in Trips \tag{22}$$

Note that the compatible trip pair, as shown in the example in Chapter 1, is the difference between number of trips and number of buses. Consider it like this, first assign a bus to every trip as none of them are compatible. At this stage, number of buses equals to number of trips. Then find all pairs of compatible trips and put them into one block that is served by one bus. Now, the number of buses is different from number of trips and the difference is the number of compatible trip pairs. Therefore, the first two terms in the objective function, number of trips minus the compatible trip pair would yield the number of buses.

The first set constraints are logistic constraints. Constraints (2) prevent the assignment of stops to the trips that have not been assigned to their respective schools. Constraints (3) assure that if a stop is assigned to a trip, that trip enters that stop. Constraints (4) enforce each PM trip starts from the school. Conservation of flow is expressed through Constraints (5). Constraints (6) are limiting the lower bound for the number of trips assigned to a school to the minimum number needed, based on the school’s population and bus capacity. Constraints (7) define the feasible trips set as the minimum number of trips plus the additional allowed trips (A). Minimum number of trips (for each school) is the ceiling integer (ceiling integer of a real number is the smallest integer that is greater than or equal to the real number) of the ratio of total number of students (to that school) divided by the bus capacity. It is the minimum number of trips, with respect to the capacity, have to be used to transport all students from their homes to the school and vice versa. The additional allowed trips is the number of trips more than the minimum number of trips (for each school) can be used. The sum of minimum number of trips and additional allowed trips is the feasible trip set for each school. Constraints (8) delete a trip from the feasible trips set if no stops are assigned to it. This is effective if additional allowed trips (A) is greater than zero.

The second set of constraints is capacity constraints. Constraints (9) are trip capacity constraints. Constraints (10) ensure that all students are served. Constraints (11) disallow the assignment of students to trips that do not pass the stop in which the students are located. Constraints (12)-(13) are flow sub-tour elimination constraints. These are formulated based on an artificial commodity which was discussed in Section 2.3.4. Constraints (14) are for eliminating symmetries. Note that if there is only one trip assigned to a school, the ID for this trip could have many values. These constraints prevent higher trip IDs from occurring before the lower ones and can speed up the search for good solutions.

The rest of the constraints are scheduling constraints. Constraints (15) calculate the travel duration of trips. Recall that SBR is an OVRP (see discussion in Chapter 2) such that the trips end at the last bus stop rather than the school. Constraints (16) are used for identifying the last stops for the trips. This last stop is the last bus stop and is the bus stop right before going back to the school. Constraints (17) calculate the end time of the trips using the start times plus the travel times. Constraints (18) are used for identifying the compatible trips. The deadheads between pairs of trips that are used for compatibility check is computed using constraints (19). Constraints (20)-(21) limit that a trip has at most one leading and one following compatible trip. Constraints (22) are the maximum ride time constraints, which ensure that the maximum travel time for trips should be less than a certain limit.

3.3 Chapter Conclusion

In this chapter, a Mixed Integer Programming model is proposed to formulate the school bus routing and scheduling problem. A few realistic assumptions are first introduced to help to define the scope of the model, which is multi-school bus routing and scheduling problem with homogeneous bus capacity. Mixed load is prohibited but multi-visit at stops is allowed. This model consists of one objective and twenty-two constraints, which are both well explained in Chapter 3.2.

Chapter 4 SCHOOL DECOMPOSITION ALGORITHM

4.1 Model Relaxation

The MIP model proposed in Chapter 3 is not efficient to be solved using an exact algorithm because of the high computation cost. Therefore, we propose a school decomposition algorithm to solve this problem. Two model relaxation techniques along with some terminologies need to be defined.

- **Trip compatibility relaxation:** Relax constraints (20) - (21) which regulate that each trip has at most one following and one leading trip.

- **Trip compatibility conversion:** trip-to-trip compatibility in constraints (18) is replaced by trip-to-school compatibility, which equals to one if a trip associated with one school can arrive at another school before its dismissal time.

- **Objective Modification:** the objective for the single-school sub-routing problem is modified as follows:

$$Min\ Z' = C_C \sum_{k \in schools} \sum_{t \in Trips\_School_k} t2s_{s,k} - \frac{M}{\left(start_t - start_{now}\right)} \times C_B \sum_{t1 \in Trips} \sum_{t2 \in Trips} b_{t1,t2} + C_T \sum_{t \in Trips} tt_t \tag{23}$$

where $start_{now}$ is the school dismissal time for current school and t is the schools for which the dismissal time are later than the current school, M is a large positive number, which is set to be 240 in the numerical experiments. The difference between the modified objective and original objective is that the modified one uses different weight on trip compatibility ($C_C > C_B$) while the original one set the weight to be equal for trip and trip compatibility ($C_C = C_B$). Trip compatibility has a higher weight for schools for which dismissal times are closer to the current school.

- **School trip capacity update:** iteratively update the maximum number of trips for each school that has not been assigned with compatible trips. It is the total number of trips used for one school (total number of potential compatible trip pair) subtracted by the actual number of compatible trip pair that has already be assigned. The initial school trip capacity equals to the minimum number of trips plus the additional allowed trips.
- **School decomposition:** divide the original multi-school problem into several single-school sub-problems and sequentially solve them.

To distinguish between the different relaxed models after the application of several relaxation techniques, we would refer to the model after trip compatibility relaxation as the ***first-degree-relaxed model***; to the model after trip compatibility relaxation and trip compatibility conversion as the ***second-degree-relaxed model***; and to the model after trip compatibility relaxation, trip compatibility conversion and objective modification as the ***third-degree-relaxed model***.

Lemma 1: The second-degree-relaxed model is a relaxation of the original SBRS problem.

Proof: trip compatibility relaxation takes off two sets of constraints regarding the trip compatibility. It enlarges the feasible region, which is a relaxation of the original problem. Recall that the original trip-to-trip compatibility is that a bus has enough time to reach the initial location for the second trip after serving the first trip. The initial locations for afternoon trips are schools. Therefore, the trip-to-school compatibility is equivalent to the trip-to-trip compatibility between. For fixed school dismissal time, trip compatibility conversion makes no changes to the model.

Lemma 2: Applying school decomposition will not alter the optimality of the second-degree-relaxed model.

Proof: The objective function and the constraints are all decomposed by the school for the second-degree-relaxed problem. It means that the summation of the optimal solution of sub-problems is the optimal solution for the non-decomposed second-degree-relaxed problem. The school decomposition does not change the optimality of the solution.

4.2 The Algorithm

The procedure for iterative school decomposition algorithm (ISDA) is described as below, and the flowchart is shown in *Figure 4*.

Step 0: Relax the original problem by trip compatibility relaxation, trip compatibility conversion, and objective modification and divide it into several single school sub-routing-problems.

Step 1: Randomly generate school sequence for consideration.

Step 2: Calculate the initial school bus trip capacity, which equals to the minimum number of trips (based on the capacity) plus the additional allowed trips.

Step 3: Solve the single-school sub-routing-problem and assign compatible trips to schools for which dismissal times are after the current school. Then update the school trip capacity.

Step 4: Repeat Step 3 and sequentially solve the single-school sub-routing-problem on each school on the list until the last school.

Step 5: Solve the blocking problem given the current solution to the school-decomposed second-degree-relaxed problem and calculate the number of buses used.

Step 6: Repeat from Step 1 to Step 5 for a finite number of iteration and find the best solution with respect to the minimum number of buses used.

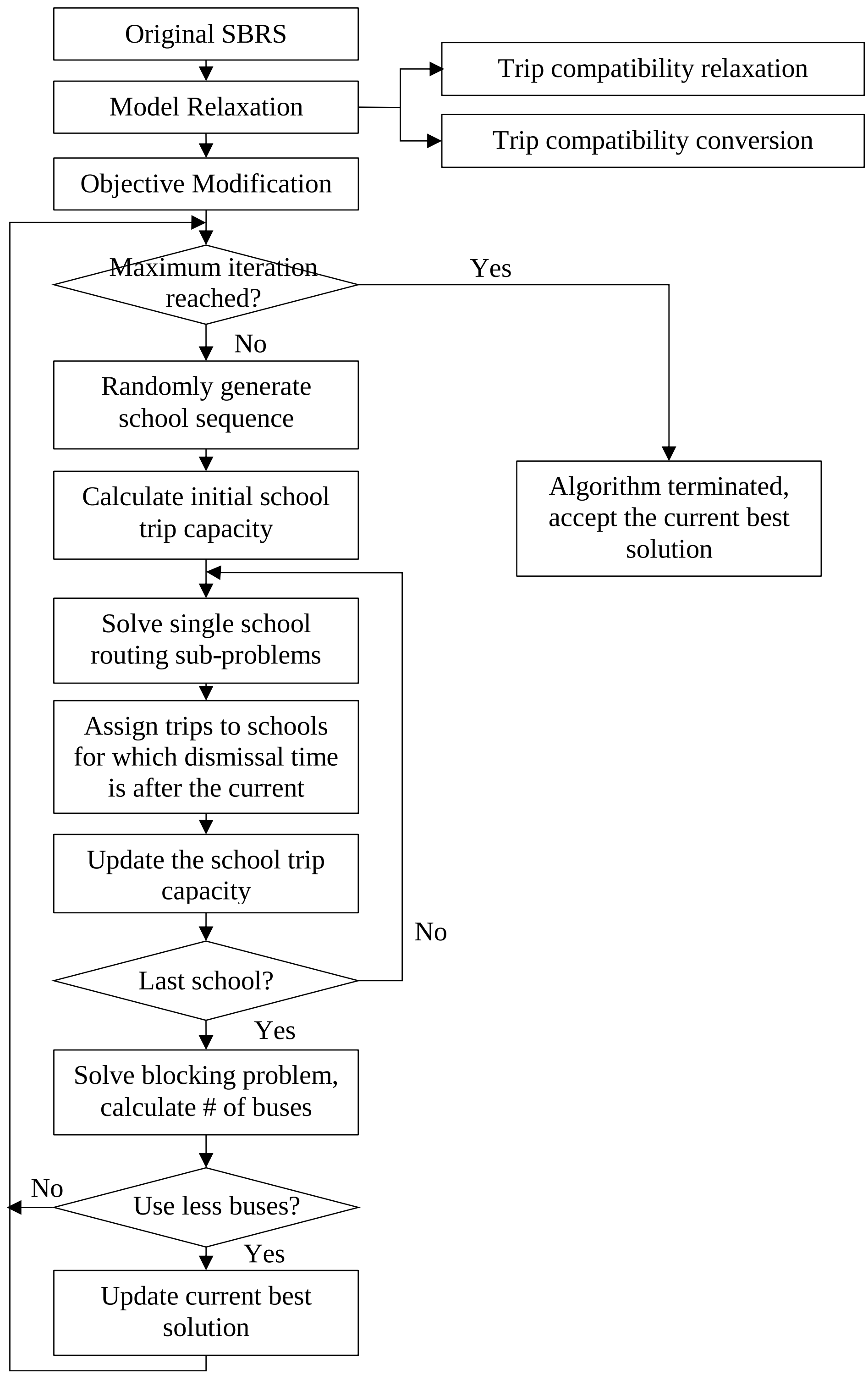

Figure 4 Flowchart of school decomposition algorithm

4.3 Discussion of the Algorithm

A few things need to be noted in the school decomposition algorithm. First, after trip compatibility relaxation, a trip compatibility over-counting problem may occur. One example of this over-counting problem is that if trip a is compatible with trips b and c but trip b and c are incompatible. In such case, one bus can serve either trip b or trip c after trip a, but not all of them. The compatible trip pair is one for original SBRS problem (notice that by constraint (20)-(21), for each trip, at most one leading and one following trip can be compatible). But the compatible trip pair will be counted as two in the second-degree-relaxed problem. Under this condition, compatible trip pair does not equal to the difference between the number of buses and the number of trips. And that will result in the increase in the actual compatible trip pair in the objective function and gives too much credit to the compatible trip pair if using the same objective function in the original SBRS problem.

Objective modification, which adjusts the weight for the trip compatibility, aims to solve the over-counting problem. Notice that the number of trips and total travel time in the objective remain the same with or without relaxation and that the decomposition does not change the optimality of the second-degree-relaxed problem. The trip compatibility relaxation will only increase the second component of compatible trips in the objective function due to the over-counting problem. The solution to the second-degree-relaxed problem is an upper bound to the original SBRS problem. By using less weight for the trip compatibility (third-degree-relaxed problem), the over-counting problem will be compensated. The third-degree-relaxed problem is an approximation of the original SBRS problem rather than a relaxation. Sensitivity analysis estimates the appropriate weight for compatibility.

Another thing is that although it was mentioned above, that trip-to-trip compatibility is equivalent to the trip-to-school compatibility; there is a relatively strong assumption to hold such one to one mapping. That is the bus service start (departure) time for all trips are equal to the school dismissal time. Any trips with a buffer after the school dismissal time is prohibited. This may not be true in reality sometimes. However, a simple approach to address this problem is to first set all bus service start time with a buffer after school dismissal time. By doing this, we find an upper bound for the original problem. After solving the problem, if the bus can reach the second trip initial stop earlier, we can manually adjust the bus service start time.

4.4 Simplified School Decomposition Algorithm

The algorithm described above, called iterative school decomposition algorithm, requires a certain number of iterations to find the best solution, which might take some time. A simplified version of this algorithm solves the problem with one single iteration. The Simplified School Decomposition Algorithm (SSDA) follows the same procedure as (ISDA) without assigning compatible trips to schools for each single-school sub-routing-problem. And therefore school trip capacity is not considered. SSDA will only try to minimize the number of trips and total travel time while maximizing the trip-to-school compatibility even if they were over-counted (See Section 4.3). Such SSDA may not find the best solution, but is more efficient and can work as a preliminary sensitivity experiment to help to find an appropriately modified weight for objective modification.

4.5 Chapter Conclusion

In this chapter, the school decomposition algorithm is proposed to solve the MIP model proposed in Chapter 3. Several techniques are first defined (in Chapter 4.1) including two model relaxation techniques (Trip Compatibility Relaxation and Trip Compatibility Conversion), one objective modification technique (Objective Modification) technique, an update procedure in the algorithm (School Trip Capacity Update) and the definition of school decomposition algorithm. Three different degree of relaxation of the original problem can be obtained by adding different relaxation techniques. The reason for applying these relaxation is to make the model solvable by the school decomposition algorithm, which is introduced in Chapter 4.2. The profound relationship, advantages and disadvantages about the relaxation techniques and algorithm are discussed in Chapter 4.3. At the end, a simplified version of school decomposition algorithm is introduced (in Chapter 4.4) to help determine the key parameters for the algorithm.

Chapter 5 CASE STUDY

Eight random mid-size problems were generated. The MIP model and the school decomposition algorithm are applied on these test problem against typical routing models that 1) minimize the number of trips and 2) minimize the total travel time. The parameters to generate the random cases are mostly based on real-world data. The detailed data structure information for each scenario are shown in Table 5. As changing the parameter of the algorithm will significantly change the algorithm's performance, different parameters set are tested on the proposed model. The algorithm parameters are defined as follows:

- **Minimum number of trips:** the smallest ceiling integer of the ratio of the total number of students for each school divided by the bus capacity. It is the minimum number of trips, with respect to the capacity, have to be used to transport all students. (e.g. school A has 180 students, the homogeneous bus capacity is 48. Then the minimum number of trips is $\lceil 180/48 \rceil = 4$).

- **Additional allowed trips (A):** the maximum additional number of trips more than the minimum number of trips can be used (e.g. same example for school A with 180 students and four minimum number of trips. If the additional allowed trips is set to be two, at most six (four plus two) trips can be used for school A).

- **Running time limit (TL):** the maximum running time for each single-school sub-routing-problem.

- **School dismissal time range:** the time difference between the earliest and latest school dismissal time. The dismissal times for the schools follow a discrete uniform distribution with 15 minutes' time intervals within that range.

Different combinations of these parameters are used to see their performance along with the impact of the data structure on the model and algorithm's performance. All the problems are solved by commercial solver FICO Mosel XPRESS on five computers with same feature: Intel® Core™ i5-2400 CPU, 3.10 GHz with 4 GB RAM. The heuristic school decomposition algorithm is coded in Python 2.7.

The total number of buses has the highest priority among all performance evaluation measurements. Besides, travel times, including travel time distribution, average travel time per trip, maximum travel time per trip and total travel time, are also important measurements of effectiveness from the level of service's perspective. For the traditional routing models (minimizing the number of trips and minimizing total travel time), the number of buses (blocks) is the output of the scheduling problem given the input trips that are the solutions of the routing model. As explained above, given routing plan with time windows, the scheduling problem becomes an assignment problem and it is easy to find its optimal solution. Therefore, the formulation of the scheduling problem given routing plan with time window is omitted.

5.1 Preliminary Experiment

5.1.1 Experiment Setup

In the preliminary experiment, we used the Simplified School Decomposition Algorithm (SSDA) and compared it to the traditional objectives. The goal of the preliminary experiment is to see whether the proposed algorithm (simplified version) can find a better solution than the traditional objectives. Another goal is to do a sensitivity analysis to find the best weight range for the modified trip compatibility coefficient in the single-school sub-routing-problems.

For each scenario, five approaches were tested: 1) minimizing the total number of trips (**MinN**); 2) minimizing the total travel time (**MinTT**); 3) minimizing the number of buses (**MinB**); 4) minimizing the number of buses as well as minimizing the total travel time (**MinB+TT**); and 5) solving the relaxed model using the simplified school-decomposition algorithm (**SSDA**) while the coefficient for trip compatibility is set to be 200 and the running time limit for each sub-problem is 5 minutes.

The first two approaches are commonly used in literature and thus will be referred as traditional models while the last three approaches were proposed in this paper. The use of the traditional models was listed in Table *2*. Minimizing the total number of trips (**MinN**) was adopted by Bodin and Berman (1979), Park, Tae and Kim (2012). Minimizing the total travel time (**MinTT**) was adopted by Bennett and Gazis (1972), Russell and Morrel (1986), Ripplinger (2005), Spada, Bierlaire, and Liebling (2005), Schittekat, Sevaux, and Sörensen (2006), Schittekat et al. (2013), Faraj et al. (2014), Kinable, Spidksma and Vandem Berghe (2014), Kang et al. (2015), Mushi, Mujuni and Ngonyani (2015), Santana, Ramiro, and Romero Carvajal (2015), Silva et al. (2015), and Yao et al. (2016).

Due to the slow rate of reduction in the gap (for exact branch and bound for MinB and MinB+TT), the solution processes for the test problems were terminated after a certain amount of running times (from 30 minutes to 24 hours). Also, the maximum ride time constraint is relaxed in this experiment because this problem is formulated for an urban school bus system where the maximum ride time is not the binding constraint.

5.1.2 Experiment Result

Table 5 shows that the proposed model MinB, MinB+TT and SSDA can find a better solution (with respect to fewer number of buses) than the traditional objectives of minimizing the number of trips or minimizing the total travel time for all test problems. The bus saving in the proposed method can go up to 30% (Scenario 7 where 26 buses from MinN can be reduced to 18 buses from SSDA). This is significant.

An important thing to notice is the relationship between MinB, MinB+TT, and SSDA. In Table 5, SSDA found the minimum number of buses in five tests (notice that scenario one has four different tests), which is the highest. MinB+TT is the second, which found the best solution four times and MinB found the best solution twice. Another comparison is the running time for these three approaches. It is safe to say that the running time for SSDA is trivial comparing to MinB+TT and MinB solved by exact branch and bound method. And the running time for the exact algorithm will go much faster than SSDA with the increase of the problem size. Overall, SSDA can find better solutions in much shorter running time than an exact algorithm.

Figure 5 shows the blocking result of each approach at all scenarios. It clearly shows that MinB, MinB+TT, and SSDA tend to have more long routes for the buses (routes with more trips). This leads to the savings in the number of buses used. The more routes with multiply trips the plan has, the more trips each bus will serve, and thus fewer buses are needed. Generally, more long routes yield to a reduction in the total number of routes (buses).

Another essential solution quality criterion is the travel time. Since, we relaxed the maximum ride time in the preliminary experiment, minimizing total travel time is

Table 5 Computational result for preliminary experiments

Scenario		**1**				**2**	**3**	**4**	**5**	**6**	**7**	**8**
# of Stops		100				200	100	100	125	100	200	200
# of Schools		20				20	20	20	25	20	20	20
①		91.4				90	121	183	90.4	91.6	89.5	91.1
②		13				16	13	13	13	13	16	14
③		0-30				0-30				0-90		0-16
④		3	2	1	0	0	0	0	0	0	0	0
MinN	**RT**	0.38	0.05	0.14	0.10	0.46	0.04	0.04	0.02	0.20	0.35	0.18
	Gap	0	0	0	0	0	0	0	0	0	0	0
	NOT	41	41	41	41	47	54	75	46	41	47	45
	NOB	**36**	**35**	**35**	**38**	**38**	**45**	**60**	**41**	**21**	**26**	**45**
MinTT	**RT**	30	180	30	30	30	60	600	360	30	60	30
	Gap	1.1	0.8	1.1	0.9	3.4	6.2	20.6	0.2	1.4	7.0	15.0
	NOT	61	59	58	41	47	54	75	46	41	47	45
	NOB	**40**	**39**	**39**	**32**	**32**	**38**	**52**	**35**	**17**	**24**	**45**
MinB	**RT**	30	180	30	60	60	60	600	360	60	60	8.5
	Gap	10.7	7.3	16.1	37.5	16.7	29.7	29.8	24.1	15.8	32.9	0.0
	NOT	43	42	47	41	47	54	75	46	41	47	45
	NOB	**31***	**31**	**32**	**30***	**30**	**36**	**45***	**36**	**16***	**24**	**44***
MinB +TT	**RT**	30	180	30	60	60	60	600	360	60	60	60
	Gap	21.1	6.7	8.4	32.8	23.1	31.1	33.2	17.8	0.08	32.6	26.6
	NOT	50	42	42	41	47	54	75	46	41	47	45
	NOB	**34**	**30***	**30**	**31**	**32**	**34***	**45***	**35**	**16***	**24**	**45**
SSDA	**RT**	10.8	6.21	2.13	0.28	3.46	5.52	25.0	3.50	0.29	17.4	4.18
	Gap	-	-	0	0	0	-	-	0	0	-	0
	NOT	65	57	49	41	47	54	75	46	41	47	45
	NOB	**42**	**34**	**28***	**31**	**28***	**39**	**46**	**33***	**17**	**18***	**45**

Note: *: Minimum number of buses among five approaches for each scenario
① Average number of student to schools
② Maximum number of stops to schools
③ School dismissal time range
④ Additional allowed trips
RT: Running time (minute)
Gap: unit in %
NOT: Number of trips (trip)
NOB: Number of buses (bus)
SSDA: Simplified school decomposition algorithm

of huge importance. *Figure 7* shows the travel time histogram, and *Figure 6* shows the total travel time. It can be seen that MinN usually has longest total travel time and has many long travel time trips (more than 40 minutes). MinTT is usually the opposite,

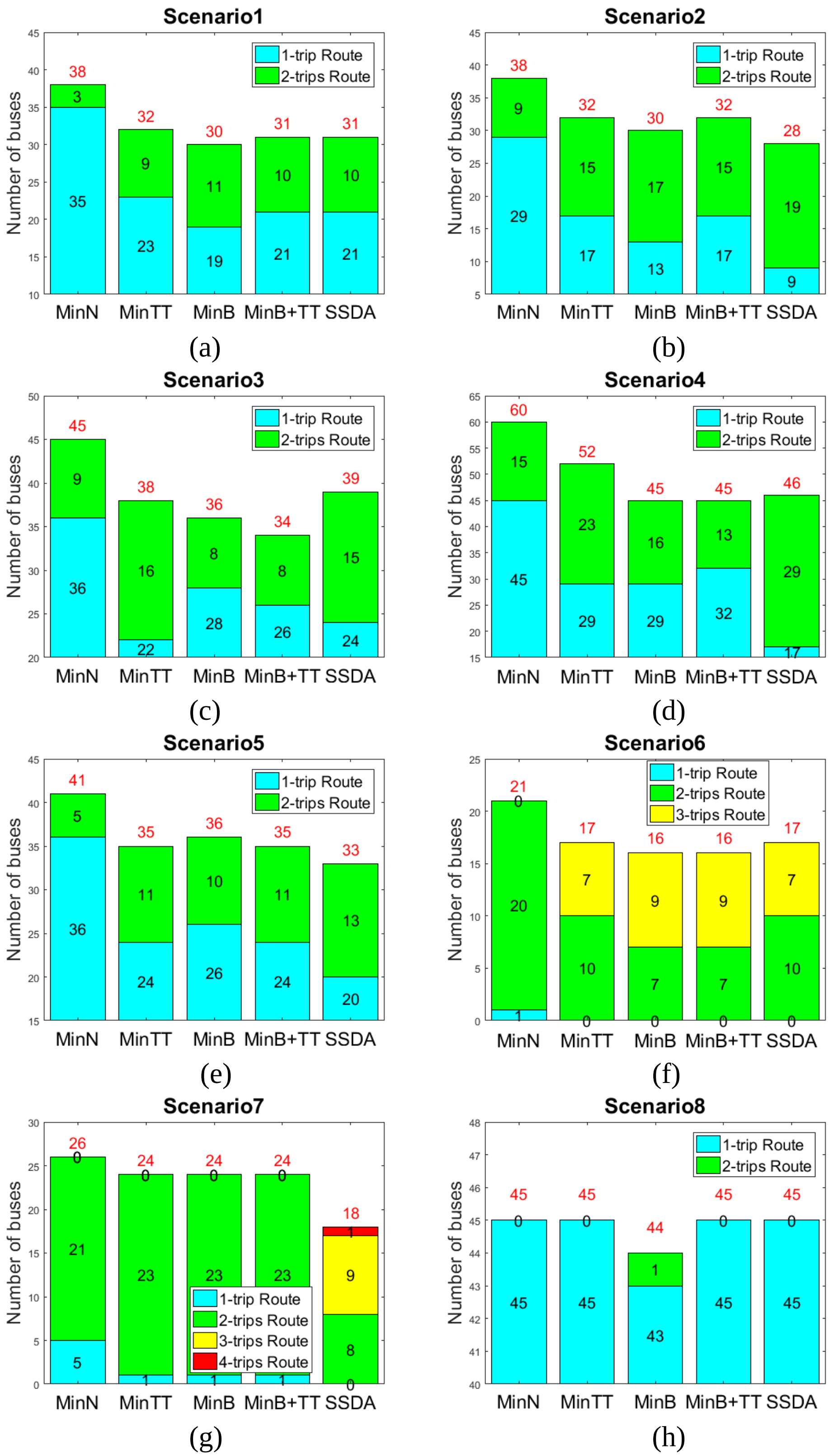

Figure 5 Summary of school bus blocking result

which has the minimum total travel time and lots of short trips. MinB+TT and SSDA both have similar good result – short trips (mostly less than 40 minutes) and short total travel time. However, MinB yields a different result from MinB+TT and SSDA. It comes from the fact that MinB only focuses on minimizing the number of buses even at the expense of extremely long trips (that could go up to 121 minutes in Scenario 4, *Figure 7*h). From this perspective, MinB+TT and SSDA make more sense.

In general, MinB+TT uses fewer buses compared to traditional models and the algorithm (SSDA) can find a better result in much shorter running time. Since we used the same data for the preliminary experiment and ISDA experiment, the more detailed analysis of the impact of data structure toward the solution quality will be discussed later.

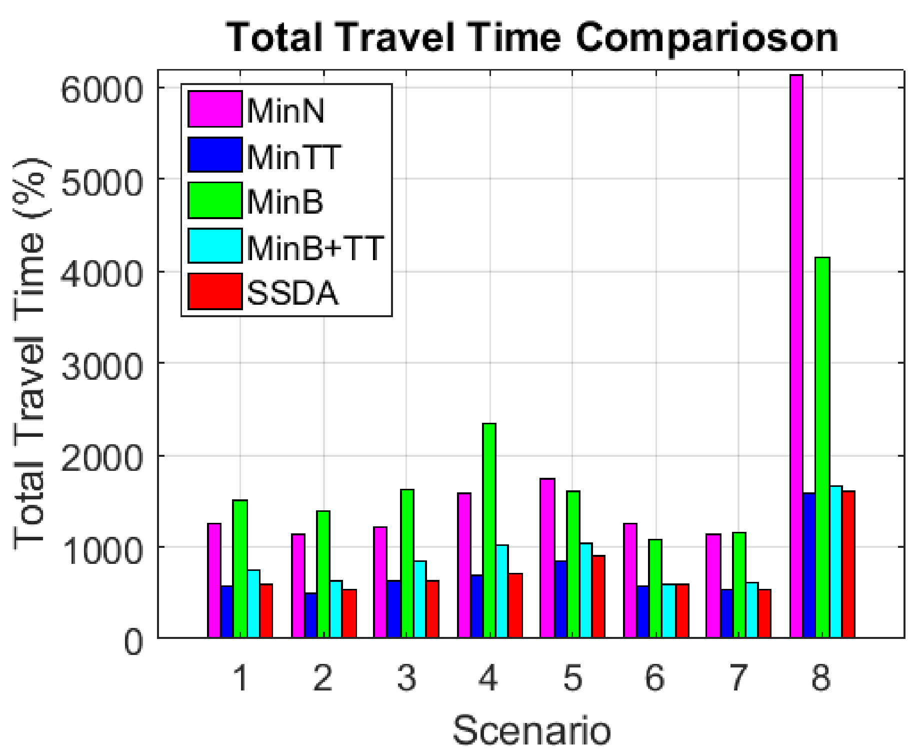

Figure 6 Total travel time comparison

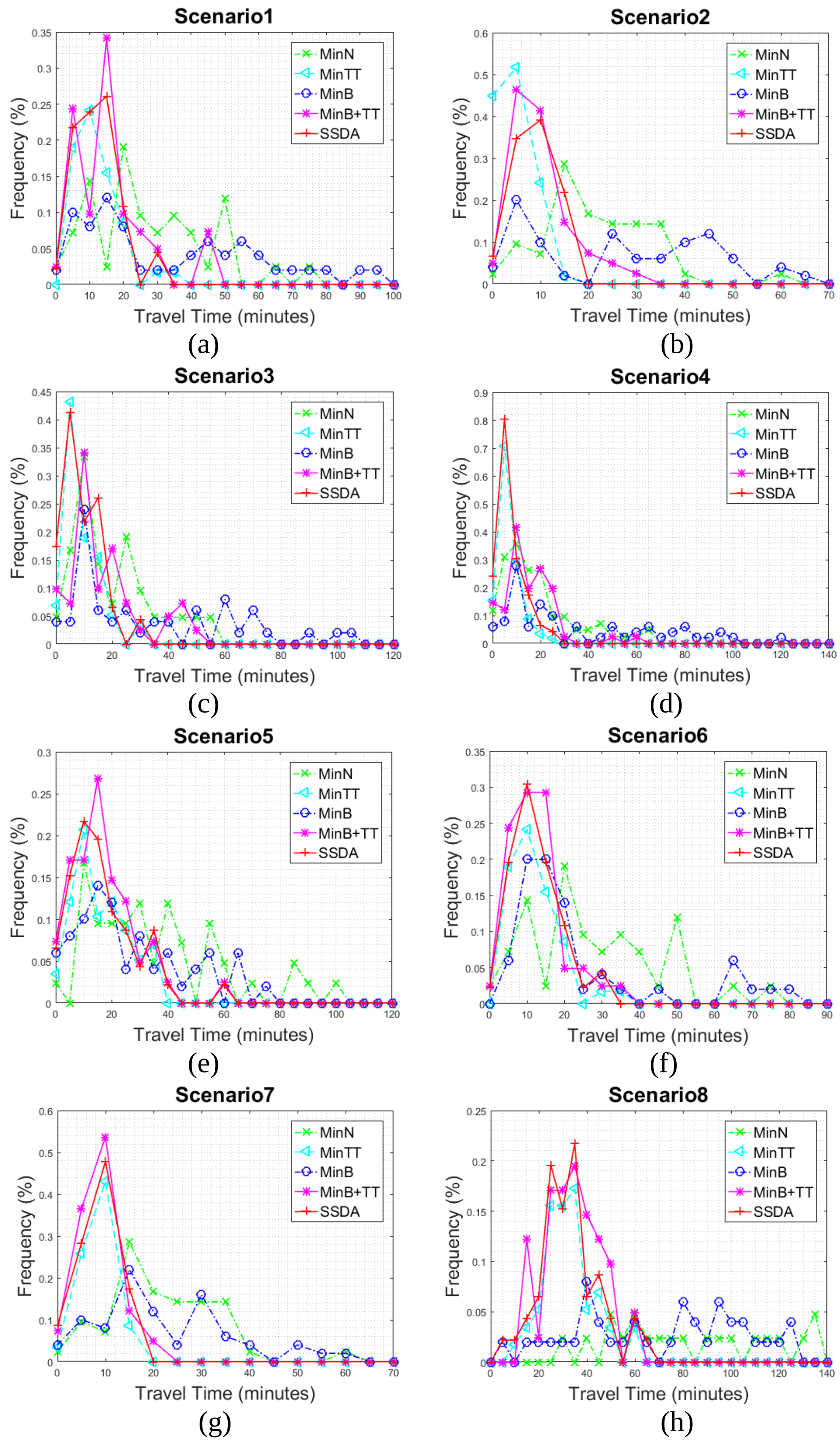

Figure 7 Travel time distribution (Frequency is calculated in 5 minutes interval and marked at the beginning of each interval)

5.1.3 Sensitivity Analysis

One important result of the preliminary experiment is to find the best weight range for the modifies objective function for single-school sub-routing-problems. The modified coefficient in the objective should compensate the over-counting problem of the trip compatibility after relaxation. If the coefficient for trip compatibility is over-emphasized, the model will try to increase the number of trips dramatically and tries to make them as compatible as possible. Once the number of trips is increased, it becomes harder to find fewer number of buses. However, if trip compatibility is under-estimated, the proposed model becomes the same as minimizing the number of trips.

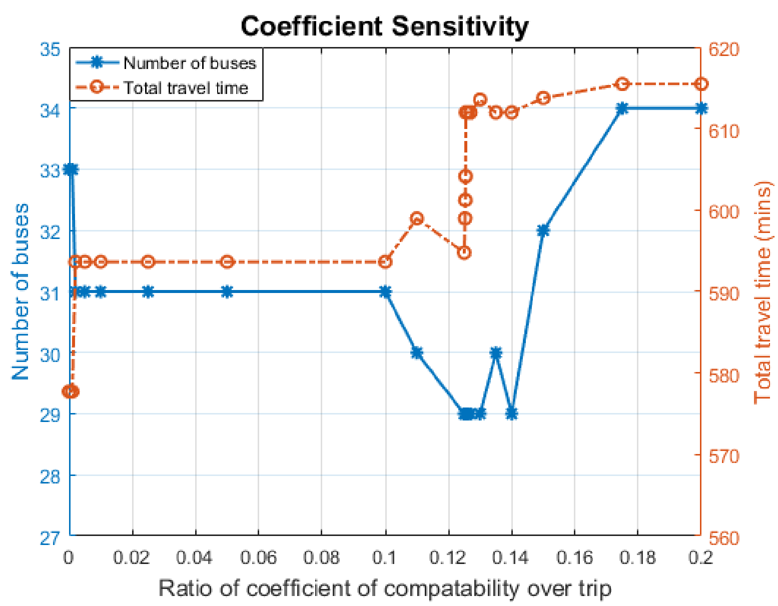

Figure 8 Sensitivity analysis on coefficient of compatibility and trips on Scenario 1 with two additional allowed trips

A sensitivity analysis is conducted based on scenario 1 (with two additional allowed trips) to find the best-recommended range for this coefficient. The coefficient for total travel time is assumed to be 1 as it is the least important among three components in the objective function. Because the ratio of coefficient of compatibility over trips is important, the coefficient for trips is fixed to be 1000, and the coefficient for compatibility is changing. *Figure 8* shows that the best-recommended ratio (coefficient of compatibility over trips) range is 0.12 - 0.14, where the best solution with the minimum number of buses and relatively small total travel time was found.

5.2 ISDA Experiment

5.2.1 Experiment Setup

In this set of experiments, we used the same eight set of randomly generated problems in the preliminary experiment and solved them using the Iterative School Decomposition Algorithm (ISDA). Eight combinations of parameters setting of ISDA are tested to find the best approach.

1) A0TL15: zero additional allowed trip and 15 seconds running time limit for each single school sub-routing-problem;

2) A0TL30: zero additional allowed trip and 30 seconds running time limit for each single school sub- routing-problem;

3) A1TL15: one additional allowed trip and 15 seconds running time limit for each single school sub- routing-problem;

4) A1TL30: one additional allowed trip and 30 seconds running time limit for each single school sub- routing-problem;

5) A1TL120: one additional allowed trip and 120 seconds running time limit for each single school sub- routing-problem;

6) A1TL30MRT: one additional allowed trip, 30 seconds running time limit for each single school sub- routing-problem and 40 minutes' maximum ride time for all trips;

7) A2TL30MRT: two additional allowed trips, 30 seconds running time limit for each single school sub- routing-problem and 40 minutes' maximum ride time for all trips;

8) A3TL30MRT: three additional allowed trips, 30 seconds running time limit for each single school sub- routing-problem and 40 minutes' maximum ride time for all trips;

The sensitivity analysis in the preliminary experiment shows that the best ratio of trip compatibility over the number of trips is 0.12-0.14. In this experiment, we used 125.364 as the coefficient for the modified trip compatibility.

5.2.2 Result Analysis

1) Problem Size

Scenario 1 is the base scenario, where there are 100 stops, 20 schools, an average of 91.4 students to each school, a maximum of 13 stop associated with one school and the school dismissal time range between 0-30 minutes (all scenarios' statistics can be found in Table *5*). In this scenarios, the minimum number of buses (NOB) is 23; 15 buses can be saved with respect to MinN and 9 buses in comparison to MinTT.

In Scenario 2, where the number of stops is increased (to 200 stops), ISDA can save 12 buses from MinN and 6 buses in comparison to MinTT. Scenario 3 is similar to scenario 1, except increasing the average number of students to 121 students per school. ISDA still finds the best solution and that 12 and 5 buses can be saved compared to that from MinN and MinTT, respectively. Scenario 4 is a further expansion of Scenario 1 and Scenario 3, which has an average of 183 students for each school. In this slightly large case, bus savings are more significant, 18 and 10 bus saving from (ISDA) with respect to MinN and MinTT. In Scenario 5, the number of stops and the

Table 6 Computational result for ISDA

Scenario		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
MinN	**NOB**	**38**	**38**	**45**	**60**	**41**	**21**	**26**	**45***
	AvgTT	30.42	24.28	22.53	21.08	37.88	30.42	24.28	136.23
	MaxTT	79.73	61.89	58.39	69.34	100.58	79.73	61.89	243.69
MinTT	**NOB**	**32**	**32**	**38**	**52**	**35**	**17**	**24**	**45***
	AvgTT	14.11	7.08	11.72	9.10	18.57	14.13	11.17	35.15
	MaxTT	35.33	15.95	33.07	29.56	39.62	35.33	17.26	64.65
SSDA	**NOB**	**31**	**28**	**39**	**46**	**33**	**17***	**18**	**45***
	AvgTT	14.48	11.38	11.85	9.58	19.49	14.61	11.27	35.43
	MaxTT	33.07	19.99	33.07	29.56	60.51	33.07	19.08	68.42
A0TL15	**NOB**	**23***	**26***	**36**	**45**	**33**	**17***	**16***	**45***
	AvgTT	18.84	11.30	11.38	9.20	19.12	14.27	11.07	35.45
	MaxTT	53.11	24.56	33.07	29.56	60.51	33.07	21.53	68.42
A0TL30	**NOB**	**23***	**26***	**36**	**44**	**33**	**17***	**16***	**45***
	AvgTT	18.84	11.49	11.41	9.23	19.12	14.27	11.10	35.45
	MaxTT	53.11	25.50	33.07	29.56	60.51	33.07	22.77	68.42
A1TL15	**NOB**	**23***	**26***	**34**	**43**	**31***	**17***	**17**	**45***
	AvgTT	19.16	10.74	10.73	9.38	18.58	14.27	10.94	35.48
	MaxTT	53.11	24.72	33.07	29.56	60.51	33.07	24.27	68.42
A1TL30	**NOB**	**23***	**26***	**35**	**42***	**31***	**17***	**18**	**45***
	AvgTT	18.01	10.60	11.01	8.96	17.35	14.33	10.45	35.44
	MaxTT	53.11	23.71	33.07	29.56	39.62	33.07	17.75	68.42
A1TL120	**NOB**	**23***	**27**	**35**	**43**	**31***	**17***	**17**	**45***
	AvgTT	18.18	10.64	11.14	9.27	18.17	14.27	10.84	35.43
	MaxTT	53.11	20.10	33.07	29.56	63.95	33.07	20.14	68.42
A1TL30 MRT	**NOB**	**28**	**26***	**35**	**42***	**31***	**17***	**17**	**50**
	AvgTT	13.41	10.45	11.30	8.74	17.76	14.34	10.92	31.40
	MaxTT	33.07	22.67	33.07	29.56	39.62	33.07	19.07	39.97
A2TL30 MRT	**NOB**	**28**	**26***	**33***	**43**	**31***	**17***	**18**	**50**
	AvgTT	13.43	10.66	10.98	8.65	17.76	14.35	10.88	31.40
	MaxTT	33.07	24.75	33.07	29.56	39.62	33.07	19.21	39.97
A3TL30 MRT	**NOB**	**27**	**26***	**33***	**43**	**31***	**17***	**17**	**50**
	AvgTT	12.60	10.64	10.67	9.03	17.76	14.41	11.01	31.40
	MaxTT	33.07	23.42	33.07	29.92	39.62	33.07	20.85	39.97

Note:
AvgTT: average travel time per trip (minutes) MaxTT: maximum travel time per trip (minutes)
*: Minimum number of buses among five approaches for each scenario

number of schools are increased a little compared to scenario 1 and the average number of student to school, and the maximum stops to school remain the same. In this scenario, ISDA found the solution with 10 and 6 fewer buses than traditional objectives of MinN and MinTT. As it can be seen, ISDA can find the better solution and use significantly fewer buses than traditional objectives of MinN and MinTT. It can be noticed that the potential saving is influenced by the problem size as well as the data structure. The larger the problem size, the more potential bus saving. The term ‘larger’ is referred to all features: the number of stops, the number of schools, the number of student to each stops, etc.

Note that trip compatibility modified coefficient is selected from the sensitivity analysis based on scenario 1. This causes scenario 1 to have relatively larger bus saving with its problem size. Therefore, in a real school bus design project, a specific sensitivity analysis for the modified objective is essential before applying ISDA. Such sensitivity analysis can help capture the unique structure of the data. SSDA is designed to conduct such sensitivity analysis effectively.

2) Additional Allowed Trips

Additional allowed trips is the key feature we used to define the potential trip set. The example in Chapter 1.2 also shows the importance of the additional allowed trip. In that example, the minimum of one trip is required to use for school 1. If the additional allowed trip is set to be zero. Only one trip can be used for school 1, then, the solution found of plan 3 will not be found since two trips are used. In the preliminary experiment, we conducted a simple illustration of the impact of “additional allowed trips” on the complexity of the problem and solution time. Four different values for the

A parameter (A = 0,1,2,3) were tested on scenario 1 (Table *5*). As it can be seen, the more additional trips are allowed, the more running time is required. However, better solutions may be found. For example, 28 buses (SSDA, A=1) is the current best solution (in Table *5*) in comparison to all other solutions from zero additional trips cases. However, the more additional trips may significantly increase the problem complexity and lead to worse solutions under the same running time.

A similar result cen be seen from Table *6* where solutions obtained from MinTT, SSDA and ISDA are listed. The percentage change is listed in Table *7*. Note the positive number (in Table *7*) means the increase percentage in comparison to the solution from MinTT while the negative number is the decrease percentage. Allowing some additional allowed trips (A > 0) will usually lead us to the minimum number of buses compared to no additional allowed trips cases. In this experiments, (A > 0) found the best solution for all scenarios except scenario 7. It will be discussed later that the school dismissal time range for scenario 7 is really large. It means that possible combinations of the compatible trips are much larger than other scenarios. Under this condition, the additional allowed trips significantly increase the feasible combinations of the compatible trips and thus booms the feasible solution set. Therefore, the algorithm may not be able to find a better solution for this much larger problem under the same small running time limit. It should be noted that in real applications, more additional trips should be allowed to try to find the best solution by simply lengthening the running time. In real applications, the running time for SBRS is less important than finding the optimal solution.

3) Different Running Time Limit

It is intuitive that the longer running time limit, the higher chance a better solution (or even the optimal solution) can be found. This is reflected in Table 6. For example, in scenario 4, the solution with 15 seconds running time limit is 45 while a better solution with 44 buses was found for 30 seconds running time limit scenario (both for A=0). For scenario 4 (A=1 cases), a better solution with 42 buses was found under 30 seconds running time limit compared to 43 buses from 15 seconds running

Table 7 Improvement percentage with respect to MinTT (%)

Scenario		**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
SSDA	**NOB**	**-3.13**	**-12.50**	**2.63**	**-11.54**	**-5.71**	**-5.56**	**-25.00**	**0.00**
	AvgTT	2.62	60.73	1.11	5.27	4.95	3.40	0.90	0.80
	MaxTT	-6.40	25.33	0.00	0.00	52.73	-6.40	-0.54	5.83
A0TL15	**NOB**	**-28.13**	**-18.75**	**-5.26**	**-13.46**	**-5.71**	**-5.56**	**-33.33**	**0.00**
	AvgTT	33.52	59.60	-2.90	1.10	2.96	0.99	-0.90	0.85
	MaxTT	50.33	53.98	0.00	0.00	52.73	-6.40	24.74	5.83
A0TL30	**NOB**	**-28.13**	**-18.75**	**-5.26**	**-15.38**	**-5.71**	**-5.56**	**-33.33**	**0.00**
	AvgTT	33.52	62.29	-2.65	1.43	2.96	0.99	-0.63	0.85
	MaxTT	50.33	59.87	0.00	0.00	52.73	-6.40	31.92	5.83
A1TL15	**NOB**	**-28.13**	**-18.75**	**-10.53**	**-17.31**	**-11.43**	**-5.56**	**-29.17**	**0.00**
	AvgTT	35.79	51.69	-8.45	3.08	0.05	0.99	-2.06	0.94
	MaxTT	50.33	54.98	0.00	0.00	52.73	-6.40	40.61	5.83
A1TL30	**NOB**	**-28.13**	**-18.75**	**-7.89**	**-19.23**	**-11.43**	**-5.56**	**-25.00**	**0.00**
	AvgTT	27.64	49.72	-6.06	-1.54	-6.57	1.42	-6.45	0.83
	MaxTT	50.33	48.65	0.00	0.00	0.00	-6.40	2.84	5.83
A1TL120	**NOB**	**-28.13**	**-15.63**	**-7.89**	**-17.31**	**-11.43**	**-5.56**	**-29.17**	**0.00**
	AvgTT	28.84	50.28	-4.95	1.87	-2.15	0.99	-2.95	0.80
	MaxTT	50.33	26.02	0.00	0.00	61.41	-6.40	16.69	5.83
A1TL30 MRT	**NOB**	**-12.50**	**-18.75**	**-7.89**	**-19.23**	**-11.43**	**-5.56**	**-29.17**	**11.11**
	AvgTT	-4.96	47.60	-3.58	-3.96	-4.36	1.49	-2.24	-10.67
	MaxTT	-6.40	42.13	0.00	0.00	0.00	-6.40	10.49	-38.17
A2TL30 MRT	**NOB**	**-12.50**	**-18.75**	**-13.16**	**-17.31**	**-11.43**	**-5.56**	**-25.00**	**11.11**
	AvgTT	-4.82	50.56	-6.31	-4.95	-4.36	1.56	-2.60	-10.67
	MaxTT	-6.40	55.17	0.00	0.00	0.00	-6.40	11.30	-38.17
A3TL30 MRT	**NOB**	**-15.63**	**-18.75**	**-13.16**	**-17.31**	**-11.43**	**-5.56**	**-29.17**	**11.11**
	AvgTT	-10.70	50.28	-8.96	-0.77	-4.36	1.98	-1.43	-10.67
	MaxTT	-6.40	46.83	0.00	1.22	0.00	-6.40	20.80	-38.17

time. However, there exist many counter examples that increasing time limit does not give a better solution but instead, has a worse solution. This might come from the randomness of the school sequence. The good part of this is that once the running time limit exceeds a threshold (15 seconds in the test scenarios), the marginal benefits of lengthening running time limit gets smaller and smaller. Therefore, in a more time demanding applications, it is better to set the running time limit close to the threshold, and it will not significantly weaken the solution quality.

4) Maximum Ride Time

The travel time is another important solution quality criterion, which has been discussed slightly in the preliminary experiment. The travel time criterion consists of four parts: travel time distribution, the maximum travel time per trip, the average travel time per trip and total travel time. Maximum ride (or travel) time, as explained earlier, is an essential constraint for the SBRS. Clearly, when maximum ride time is not the binding constraint, the solution is the same with or without maximum ride constraint. It can be seen that even without the maximum ride time constraint, much of the maximum ride time is still within 40 minutes (Table *6*). This is accomplished by the fact that minimizing total travel time is also included in the objective even though the weight is small. However, there are certain situations in which merely minimizing total travel time in the objective is not enough. Thus we need to incorporate maximum ride time constraint. When it becomes the binding constraint, adding this constraint would significantly change the solution structure. For example, in scenario 8, without maximum ride time constraint, 45 buses can accommodate the school transportation demand. However, the best solution with 45 buses would still use trips as long as 68 minutes, which is too long for students. By adding the maximum ride time constraint, the maximum travel time for trips are significantly reduced (less than 40 minutes) but the number of buses increases to 50.

The mean of the travel time is an essential criterion to describe the travel time distribution. The average travel time is also listed in Table *6*. It can be seen that the average travel time for ISDA is usually pretty small, less than 20 minutes. The underlying reason is that shorter trips are easier to be compatible with other trips and that minimizing total travel time is also in the objective. Therefore, ISDA would tend to form short trips, which is a good practice for school bus problem.

5) Tradeoff Between Bus Saving and Travel Time Increase

This gain in savings in the number of buses needed for serving the trips does not come free. Table *7* listed the change percentage of all tests compared to MinTT, which is the best in traditional models (MinTT & MinN). *Figure 9* shows the tradeoff between the reduction in the number of buses and the increase in the average travel time per bus and the maximum ride time per bus. The comparison is made between the best results from the proposed models (ISDA) and the best results from traditional models (MinTT & MinN).

It can be seen that bus saving usually is achieved at the expense of an increase in average and maximum travel time. Take scenario 1 for instance, A1TL30 would reduce the NOB by 28.13% compared to MinTT at the expense of 27.64% increase in average travel time and 50.33% increase in the maximum ride time. When limiting maximum ride time, the NOB saving reduces to 12.50% (A1TL30MRT), but the increase in average travel time and maximum travel time also drops to 4.82% and 6.4%,

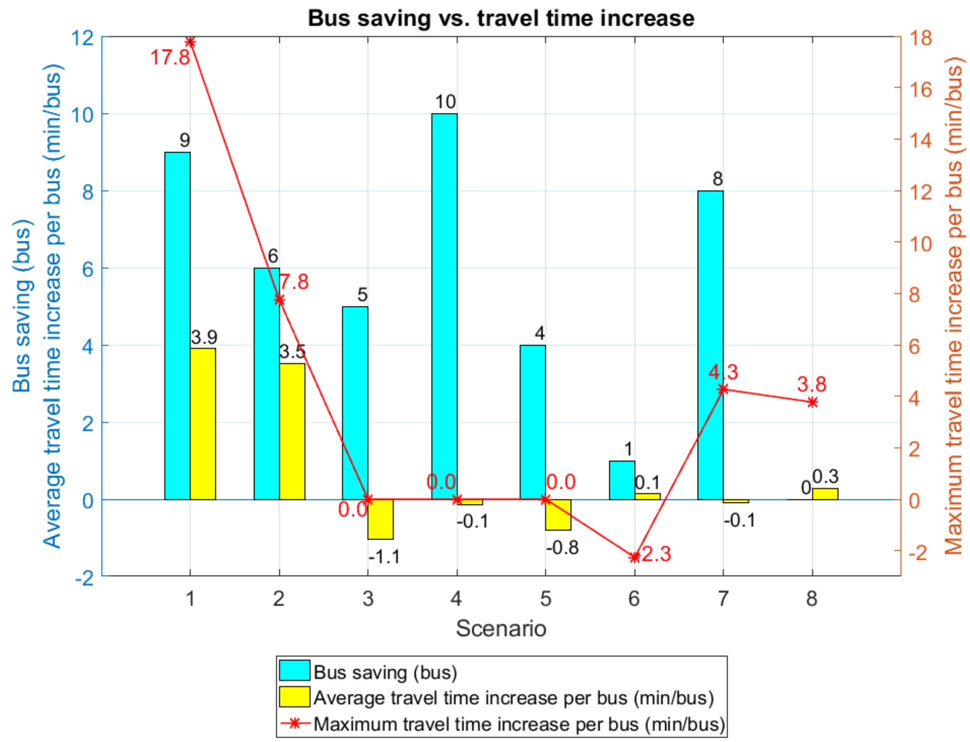

Figure 9 Tradeoff between bus saving and travel time increase
from best new models and best traditional models
(Negative travel time increase means the decrease of travel time)

respectively. A similar result can be seen in other scenarios. The overall trend is that the less bus required, the higher average and maximum travel time. But remember that the annual cost for a school bus and a driver is roughly $50,000 – $100,000 while the travel time increase is trivial compared to the bus saving. From a financial point of view, the savings gained by using fewer buses could easily justify the additional travel times. There are indeed some scenarios that ISDA find solutions that use fewer buses and require less average and maximum ride time. For example, A1TL30 found such result in scenario 3 and 5 compared to MinTT. It shows the proposed algorithm can find much better result than traditional objectives with respect to all criteria (less number of buses and less average and maximum ride time).

6) School Dismissal Time Range

Scenario 6 to Scenario 8 examine the performance of the model under different school dismissal time ranges. If the range is too long, trips can easily be compatible with each other (scenario 6 and scenario 7), on the other hand, if the range is too short, it might be impossible for trips to be compatible (scenario 8). In scenario 6, where the school dismissal time is set to be 90 minutes, only four buses and one bus can be saved from MinN and MinTT, respectively (Table *5*, Table *6*). Eight and ten buses were saved compared to MinN and MinTT in scenario 7. And there is no bus saving in scenario 8. The cross-examination reveals that the proposed model cannot have better results than the traditional models under the extremely small school dismissal time range. It clearly shows that the data structure has a remarkable impact on the model and algorithm's performance and solution quality.

7) Upper bound and pseudo-reduction

It is well known that for a minimization problem, any feasible solution is an upper bound to the primal problem. In school bus routing and scheduling problem, one simple upper bound is the number of trips. The rationale is that assigning each trip with one bus, the number of trips equals to the number of buses. Clearly, it is a feasible solution, and thus is an upper bound. Based on this idea, a loose upper bound (LUB) is the summation of minimum number of trips for each school based on capacity constraint over all schools plus the product of additional allowed trips (per school) and number of schools. Such loose upper bound can be strengthened (SUB) by the actual minimum number of trips found by different models (MinN, MinTT, MinB, MinB+TT, SSDA, ISDA etc.). When additional allowed trip is zero, loose upper bound equals to the strengthened upper bound.

Notice that upper bound also gives some insight about the complexity (or the size) of the problem, the larger the upper bound, the more trips need to be used, and the harder the problem is. Therefore, we define the pseudo-reduction (PR) as follows:

$$PR = \left(1 - \frac{number\ of\ buses}{strengthened\ upper\ bound}\right) \times 100\% \quad (24)$$

Table 8 Pseudo-reduction of different approaches (%)

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
MinN	7.32	19.15	16.67	20.00	10.87	48.78	44.68	10.00
MinTT	21.95	31.91	29.63	30.67	23.91	99.59	48.94	10.00
MinB	26.83	36.17	33.33	40.00	21.74	99.61	48.94	12.00
MinB+TT	24.39	31.91	37.04	40.00	23.91	99.61	48.94	10.00
SSDA	24.39	40.43	27.78	38.67	28.26	99.59	61.70	10.00
ISDA	43.90	44.68	37.04	44.00	32.61	99.59	65.96	10.00
ISDA_MRT	58.54	44.68	38.89	44.00	32.61	99.59	63.83	0.00

Note:
ISDA: Pseudo-reduction of the best solution found by ISDA without MRT
ISDA_MRT: Pseudo-reduction of the best solution found by ISDA with MRT

It can be seen that the maximum pseudo reduction is 99%, which implies (almost for sure) that all the buses will service more than one trip. Without maximum ride time (MRT) constraint (for Scenario 8), the minimum pseudo reduction is 10%, which means (assuming only 1- /2-trip(s) routes exist) 20% of trips are compatible with other trips can thus can be served by one bus. This, again, shows the importance of the trip compatibility.

8) Computational Time

The computational time depicts the efficiency of the proposed model and algorithm. The general idea of the computational time is the how would the running time change with respect to the increase of the problem size. However, the data structure for SBRS is complicated and many data properties contribute the complexity of the problem – number of school, number of stops, total students, the distribution of students at each school and each stop, the geometric distribution of stops with respect to school etc. Due to the complexity of the problem, finding a strong indicator to show the problem complexity is hard enough. The computational time is shown in Figure *10*, where computational time is not a strict increasing line with respect to any single problem complexity indicator (Number of stops, total number of students, number of trips or minimum number of buses). This demonstrates the complexity of the problem and the impact of data structure on the performance (efficiency) of the model and

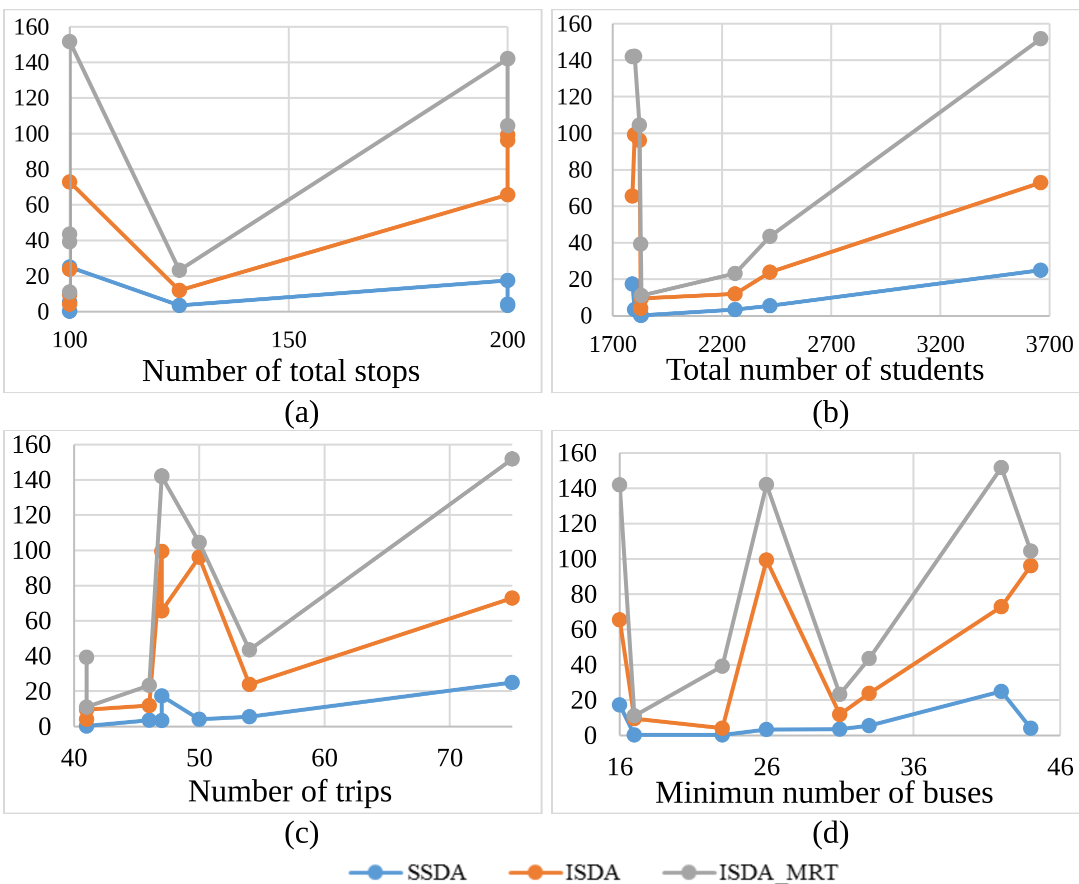

Figure 10 Computational Time

Note:
SSDA: Computation time of simplified school decomposition algorithm
ISDA: Computation time of the ISDA without MRT
ISDA_MRT: Computation time of the ISDA with MRT

algorithm. However, the overall trend (especially from Figure *10*(b) and (c)) is that ISDA with maximum ride time is the most time consuming algorithm while SSDA is the most time efficient algorithm. Also, the running time for ISDA with MRT increases much faster than ISDA and SSDA with respect to the increase of the "problem size". The problem size is referred to the overall complexity of the problem, which is affected by the number of students, number of stops and other data structure.

5.3 Chapter Conclusion

In this chapter, the proposed model and school decomposition algorithm are tested on eight set of randomly generated problems again the traditional models. The result shows that the proposed model and algorithm found better result in terms of minimum number of buses (up to 30% fewer number of buses comparing to existing methods) in a reasonable amount of running time. From the travel time's perspective, the proposed model and algorithm also find solutions with relatively low average and maximum travel time for each trip. A preliminary experiment is first conducted (in Chapter 5.1) on simplified school decomposition algorithm to determine the key parameter – adjusted weight for trip compatibility in Objective Modification. Then a full version of the experiment is tested (in Chapter 5.2) on Iterative School Decomposition Algorithm (which is the full version of the school decomposition algorithm) using the parameters found in the preliminary experiment. The sensitivity analysis on several data structure parameters (number of schools/stops, school dismissal time range etc.) and algorithm parameters (additional allowed trips, running time limit, etc.) are conducted to demonstrate the impact of these parameters toward the performance of the algorithm. An upper bound based on the number of trips is proposed, which give some insights about the complexity of the problem. The improvement of the solution found by the algorithm is analyzed in comparison to this upper bound. The computational time shows that the complexity of the problem and that efficiency of the algorithms along with the running time increase rate with respect to the increase of the problem size.

Chapter 6 CONCLUSION

6.1 Summary

In this thesis, we proposed a MIP model and school decomposition algorithm to solve the multi-school homogeneous-fleet school bus routing and scheduling problem (SBRS) problem.

In Chapter 1, we first defined a few terminologies and used them to distinguish among several related problems: school bus routing problem, school bus blocking/scheduling problem, and school bus routing problem with time window constraints. We shed light on the trip compatibility. It is the underlying reason why separating routing and scheduling problem will yield worse solution than solving them simultaneously. The trip compatibility is, therefore, the key concept in this thesis that connects routing and scheduling problems together.

In Chapter 2 we summarized the existing work on school bus routing and scheduling problem. The different classification and different objective of SBRS were summarized along with six families of constraints for formulating SBRS problem. Formulations and algorithms for SBR and SBRS were discussed in detailed with their strengths, weaknesses, and applications. Both exact algorithm and heuristics were discussed. The research gap is discussed, which can be expected to be filled by this thesis.

A mathematical MIP model for SBRS was presented in Chapter 3. The assumptions of the model and some strategies to relax parts of the assumptions were also discussed. The objective and all constraints were explained in detailed.

In Chapter 4, the School Decomposition Algorithm (SDA) was proposed to solve the mathematical model proposed in Chapter 3. Model relaxation and modification were applied to make the model suitable for the SDA. The algorithm procedure and the rationale behind this algorithm were discussed. Two types of SDA were designed for the full version experiment and preliminary experiment.

Chapter 5 presented the computation experiments where eight sets of random generated mid-size problems were used to test the performance of the proposed model and algorithm in comparison to the traditional models. It was shown that the proposed model can find better solutions with respect to the minimum number of buses and relatively small travel time and that the SDA is an efficient and effective way to solve the model. The proposed model and algorithm can find better result with respect to fewer number of buses than existing methods for all test scenarios. Such bus saving can go up to 30%. At some scenarios, the proposed model and algorithm beat the traditional methods not only with fewer number of buses but also smaller average and maximum travel time per trip. Overall, significant financial benefits can be obtained by school bus operator by applying the proposed model and algorithm.

6.2 Future Research

Several topics are still open to future research. One of them is the performance of the model and algorithm on large real-world instances. The problem size for real world school bus routing and scheduling is usually larger than those used in the computational experiments in this research. To solve such problems in a reasonable amount of time, more efficient models and algorithms are required. Benchmark problems can also be solved to compare the model and algorithm's performance.

The second potential research direction is developing an algorithm to solve each single-school sub-routing-problem in SDA. In this thesis, we used commercial solver FICO Xpress to solve each single-school problems. Many heuristics including insertion method, sweep method, etc. as discussed in the literature review might be incorporated into the SDA framework. These efficient heuristics may help SDA to handle much larger problems more efficiently.

The third part is associated with blocking problem. In this thesis, we only considered minimizing the number of buses, but there are other solution quality criteria. For example, the balance of travel time for each bus, the distance between initial and end location for each bus, etc. Without these constraints, a solution may have extremely unbalanced travel time for buses where one bus only accommodates a short trip while another bus needs to service several long trips. Another unwanted solution is that a bus ends at the final location that is far away from its depot such that it needs to travel almost the same distance to go back to the depot. It is a good practice to include these criteria into the model.

Moreover, the school dismissal time is set to be fixed in this thesis due to the need for converting from trip-to-trip compatibility to trip-to-school compatibility. But if we allow the trips to start after the school dismissal time with a buffer, more compatible trips may occur, and thus fewer buses are needed. In the end, this thesis dealt only with the school bus routing and scheduling problem. The first step of bus stop generation is not considered in this thesis. It is interesting to see if augmenting the school bus stop generation and school bell time adjustment would yield better solutions within a realistic solution time.

Bibliography

[1] Angel, R.D., Caudle, W.L., Noonan, R., Whinston, A., 1972. Computer-assisted school bus scheduling. Management Science 18 (6), 279–288.

[2] Babin, G., Deneault, S. and Laporte, G., 2007. Improvements to the Or-opt heuristic for the symmetric travelling salesman problem. Journal of the Operational Research Society, 58(3), pp.402-407.

[3] Badeau, Philippe, et al. "A parallel tabu search heuristic for the vehicle routing problem with time windows." Transportation Research Part C: Emerging Technologies 5.2 (1997): 109-122.

[4] Balinski, M., and Quandt, R. (1964), "On an integer program for a delivery problem," Operations Research 12, 300-304.

[5] Beasley JE. 1983. Route-first cluster-second methods for vehicle routing. Omega;11: 403–8.

[6] Bektaş, T. and Elmastaş, S., 2007. Solving school bus routing problems through integer programming. Journal of the Operational Research Society, 58(12), pp.1599-1604.

[7] Bennett, B.T. and Gazis, D.C., 1972. School bus routing by computer. Transportation Research, 6(4), pp.317-325.

[8] Bodin, L.D., Berman, L., 1979. Routing and scheduling of school buses by computer. Transportation Science 13 (2), 113–129.

[9] Bögl, M., Doerner, K.F. and Parragh, S.N., 2015. The school bus routing and scheduling problem with transfers. Networks, 65(2), pp.180-203.

[10] Bowerman, R., Hall, B., and Calamai, P., 1995. A multi-objective optimization approach to urban school bus routing: Formulation and solution method. Transportation Research Part A: Policy and Practice, 29(2), pp.107-123.

[11] Braca, J., et al., 1994. A Computerized Approach to the New York City School Bus Routing Problem.

[12] Bramel, J. and Simchi-Levi, D. (1992) A location based heuristic for general routing problems. Working Paper, Department of Industrial Engineering and Operations Research, Columbia University.

[13] Brandao, J., and Mercer, A. "A tabu search algorithm for the multi-trip vehicle routing and scheduling problem." European journal of operational research 100.1 (1997): 180-191.

[14] Caceres, H., Batta, R., and He, Q. "School Bus Routing with Stochastic Demand and Duration Constraints." submitted to Transportation Science (2014).

[15] Chapleau, L., Ferland, J.A., Rousseau, J.-M., 1985. Clustering for routing in densely populated areas. European Journal of Operational Research 20 (1), 48–57.

[16] Chen, D.S., et al., 1990. A bus routing system for rural school districts. Computers & Industrial Engineering, 19(1-4), pp.322-325.

[17] Chen, X., et al., 2015. Exact and metaheuristic approaches for a bi-objective school bus scheduling problem. PloS one, 10(7), p.e0132600.

[18] Christofides, N., Mingozzi, A. and Toth, P., 1979. Vehicle routing problem. In N. Combinatorial Optimization. Wiley. Chichester, 313-338

[19] Christofides, N., Mingozzi, A. and Toth, P., 1981(a). Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations. Mathematical Programming, 20(1), pp.255-282.
[20] Christofides, N., Mingozzi, A. and Toth, P., 1981(b). State-space relaxation procedures for the computation of bounds to routing problems. Networks,11(2), pp.145-164.
[21] Corberán, A., et al., R.A.F.A.E.L., 2002. Heuristic solutions to the problem of routing school buses with multiple objectives. Journal of the operational research society, 53(4), pp.427-435.
[22] de Souza Lima, F.M., et al., 2017. A multi-objective capacitated rural school bus routing problem with heterogeneous fleet and mixed loads. 4OR, pp.1-28.
[23] Desrochers, M., Desrosiers, J., and Solomon, M., 1992. A new optimization algorithm for the vehicle routing problem with time windows. Operations Research, 40(2), pp.342-354.
[24] Desrochers, M., Soumis, F. 1988. A Generalized Permanent Labeling algorithm for the shortest path problem with time windows. INFOR 26, 191-192
[25] Desrosiers, J., Dumas, Y. and Soumis, F., 1986. A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows. American Journal of Mathematical and Management Sciences, 6(3-4), pp.301-325.
[26] Desrosiers, J., et al. "Time constrained routing and scheduling." Handbooks in operations research and management science 8 (1995): 35-139.
[27] Desrosiers, J., et al., 1981. An overview of a school busing system. In: Jaiswal, N.K. (Ed.), Scientific Management of Transport Systems. North-Holland, Amsterdam, pp. 235–243.
[28] Desrosiers, J., et al., 1986. TRANSCOL: a multi-period school bus routing and scheduling system. TIMS Studies in the Management Sciences 22, 47–71.
[29] Desrosiers, J., Sauvé, M., and Soumis, F., 1988. Lagrangian relaxation methods for solving the minimum fleet size multiple traveling salesman problem with time windows. Management Science, 34(8), pp.1005-1022.
[30] Desrosiers, J., Soumis, F. and Desrochers, M., 1984. Routing with time windows by column generation. Networks, 14(4), pp.545-565.
[31] Desrosiers, J., Soumis,F., and Desrochers, M. "Routing with time windows by column generation." Networks 14.4 (1984): 545-565.
[32] Díaz-Parra, O., et al., 2012, November. A vertical transfer algorithm for the school bus routing problem. In Nature and Biologically Inspired Computing (NaBIC), 2012 Fourth World Congress on (pp. 66-71). IEEE.
[33] Dulac, G., Ferland, J.A., Forgues, P.A., 1980. School bus routes generator in urban surroundings. Computers and Operations Research 7 (3), 199–213.
[34] Faraj, M.F., et al., 2014, October. A real geographical application for the school bus routing problem. In Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on (pp. 2762-2767). IEEE.
[35] Fisher, M.L., and Jaikumar, R., 1981. A generalized assignment heuristic for vehicle routing. Networks, 11(2), pp.109-124.
[36] Fügenschuh, A., 2009. Solving a school bus scheduling problem with integer programming. European Journal of Operational Research, 193(3), pp.867-884.

[37] Gavish, B., Shlifer, E., 1979. An approach for solving a class of transportation scheduling problems. European Journal of Operational Research 3 (2), 122–134.
[38] Gendreau M, Laporte G, Potvin, J-Y. Metaheuristics for the vehicle routing problem. GERAD research report G-98-52, Montreal, Canada, 1998.
[39] Gendreau, M., et al., A., 2008. Metaheuristics for the vehicle routing problem and its extensions: A categorized bibliography. In The vehicle routing problem: latest advances and new challenges (pp. 143-169). Springer US.
[40] Gendreau, M., Hertz, A. and Laporte, G. 1992. "New insertion and postoptimization procedures for the traveling salesman problem." Operations Research 40.6 (1992): 1086-1094.
[41] Gendreau, M., Hertz, A. and Laporte, G., 1994. A tabu search heuristic for the vehicle routing problem. Management Science, 40(10), pp.1276-1290.
[42] Gillett, B.E. and Johnson, J.G., 1976. Multi-terminal vehicle-dispatch algorithm. Omega, 4(6), pp.711-718.
[43] Gillett, B.E., and Miller, L.R., 1974. A heuristic algorithm for the vehicle dispatch problem. Operations Research, 22(2), pp.340-349.
[44] Glover, F., 1989. Tabu search-part I. ORSA Journal on Computing, 1(3), pp.190-206.
[45] Golden, B.L., et al., 1998. The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results. In Fleet management and logistics (pp. 33-56).
[46] Hargroves, B.T., Demetsky, M.J., 1981. A computer-assisted school bus routing strategy: a case study. Socio-Economic Planning Sciences 15 (6), 341–345.
[47] Held, M. and Karp, R.M., 1970. The traveling-salesman problem and minimum spanning trees. Operations Research, 18(6), pp.1138-1162.
[48] Kang, M., et al., 2015. Development of a genetic algorithm for the school bus routing problem. International Journal of Software Engineering and Its Applications, 9(5), pp.107-126.
[49] Kim, B.I., Kim, S. and Park, J., 2012. A school bus scheduling problem. European Journal of Operational Research, 218(2), pp.577-585.
[50] Kinable, J., Spieksma, F.C.R. and Vanden Berghe, G., 2014. School bus routing—a column generation approach. International Transactions in Operational Research, 21(3), pp.453-478.
[51] Kontoravdis, G., and Bard, J.F. "A GRASP for the vehicle routing problem with time windows." ORSA Journal on Computing 7.1 (1995): 10-23.
[52] Kumar, Y. and Jain, S., 2015, September. School bus routing based on branch and bound approach. In Computer, Communication and Control (IC4), 2015 International Conference on (pp. 1-4). IEEE.
[53] Lamatsch, A. "An approach to vehicle scheduling with depot capacity constraints." Computer-Aided Transit Scheduling. Springer Berlin Heidelberg, 1992. 181-195.
[54] Laporte, G., 1992. The vehicle routing problem: An overview of exact and approximate algorithms. European Journal of Operational Research, 59(3), pp.345-358.
[55] Laporte, G., Mercure, H., and Nobert, Y. (1986), "An exact algorithm for the asymmetrical capacitated vehicle routing problem," Networks 16, 33-46.

[56] Laporte, G., Nobert, Y. and Arpin, D., 1986. An exact algorithm for solving a capacitated location-routing problem. Annals of Operations Research, 6(9), pp.291-310.
[57] Laporte, G., Nobert, Y. and Taillefer, S., 1988. Solving a family of multi-depot vehicle routing and location-routing problems. Transportation Science, 22(3), pp.161-172.
[58] Lenstra, J.K. and Kan, A.R., 1975. Some simple applications of the travelling salesman problem. Journal of the Operational Research Society, 26(4), pp.717-733.
[59] Li, L.Y.O., and Fu, Z., 2002. The school bus routing problem: a case study. Journal of the Operational Research Society, 53(5), pp.552-558.
[60] Lin, S., Computer solutions of the traveling salesman problem, Bell System Technical Journal, 1965; 44:2245-2269.
[61] Mesquita, M., and Paixão, J. "Multiple depot vehicle scheduling problem: A new heuristic based on quasi-assignment algorithms." Computer-Aided Transit Scheduling. Springer Berlin Heidelberg, 1992. 167-180.
[62] Miller, C., A. Tucker and R. Zemlin. 1960. Integer Programming Formulations and Traveling Sales- man Problems. J. Assoc. Computer. Mach. 7, 326-329.
[63] Mushi, A.R., Mujuni, E. and Ngonyani, B., 2015. Optimizing Schedules for School Bus Routing Problem: the case of Dar Es Salaam Schools. International Journal of Advanced Research in Computer Science, 6(1).
[64] National Association of State Directors of Pupil Transportation Services.1999. School Bus Seat Capacity
[65] Newton, R.M. and Thomas, W.H., 1974. Bus routing in a multi-school system. Computers & Operations Research, 1(2), pp.213-222.
[66] Newton, R.M., Thomas, W.H., 1969. Design of school bus routes by computer. Socio-Economic Planning Sciences 3 (1), 75–85.
[67] Or, I., Traveling Salesman-Type Combinatorial Problems and Their Relation to the Logistics of Regional Blood Banking. Ph.D. Thesis, Department of Industrial Engineering and Management Sciences, Northwestern University, Evanston, IL, 1976.
[68] Osman, I.H., 1993. Meta-strategy simulated annealing and Tabu search algorithms for the vehicle routing problem. Annals of operations research,41(4), pp.421-451.
[69] Pacheco, J. and Martí, R., 2006. Tabu search for a multi-objective routing problem. Journal of the Operational Research Society, 57(1), pp.29-37.
[70] Park, J., and Kim, B.I., 2010. The school bus routing problem: A review. European Journal of Operational Research, 202(2), pp.311-319.
[71] Park, J., Tae, H. and Kim, B.-I., 2012. A post-improvement procedure for the mixed load school bus routing problem. European Journal of Operational Research, 217(1), pp.204-213.
[72] Potvin, J.-Y., and J.-M. Rousseau. "A parallel route building algorithm for the vehicle routing and scheduling problem with time windows." European Journal of Operational Research 66.3 (1993): 331-340.

[73] Potvin, J.-Y., T. Kervahut and J.-M. Rousseau (1992). A Tabu search heuristic for the vehicle routing problem with time windows. Publication 855, Centre de Recherche sur les Transports, Universite de Montreal.
[74] Prins, C., 2004. A simple and effective evolutionary algorithm for the vehicle routing problem. Computers & Operations Research, 31(12), pp.1985-2002.
[75] Renaud, J., Laporte, G., and Boctor, F.F., 1996. A tabu search heuristic for the multi-depot vehicle routing problem. Computers & Operations Research,23(3), pp.229-235.
[76] Ripplinger, D., 2005. Rural school vehicle routing problem. Transportation Research Record: Journal of the Transportation Research Board, (1922), pp.105-110.
[77] Rochat, Y., and Semet, F. (1994), "A tabu search approach for delivering pet food and flour in Switzerland," Journal of the Operational Research Society 45, 1233-1246.
[78] Russell, R.A., and Morrel, R.B., 1986. Routing special-education school buses. Interfaces, 16(5), pp.56-64.
[79] Santana, L., Ramiro, E. and Romero Carvajal, J.D.J., 2015. A hybrid column generation and clustering approach to the school bus routing problem with time windows. Ingeniería, 20(1), pp.101-117.
[80] Savas E. 1978. On equity in providing public services. Management Sci., 24, pp.800-808
[81] Schittekat, P., Sevaux, M., Sörensen, K., 2006. A mathematical formulation for a school bus routing problem. In: Proceedings of the IEEE 2006 International Conference on Service Systems and Service Management, Troyes, France.
[82] Silva, C.M., et al., 2015, September. A Mixed Load Solution for the Rural School Bus Routing Problem. In Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on (pp. 1940-1945). IEEE.
[83] Simchi-Levi, D. and Bramel, J. (1990) On the optimal solution value of the capacitated vehicle routing problem with unsplit demands. Working Paper, Department of IE & OR, Columbia University, New York.
[84] Solomon, M.M., 1987. Algorithms for the vehicle routing and scheduling problems with time window constraints. Operations Research, 35(2), pp.254-265.
[85] Spada, M., Bierlaire, M., Liebling, Th.M., 2005. Decision-aiding methodology for the school bus routing and scheduling problem. Transportation Science 39, 477–490.
[86] Syslo MM, Deo N and Kowalik JS (1983). Discrete Optimization Algorithms with Pascal Programs. Prentice-Hall Inc.: New Jersey.
[87] Taillard, É., et al. "A tabu search heuristic for the vehicle routing problem with soft time windows." Transportation Science 31.2 (1997): 170-186.
[88] Thangiah, S.R., and Nygard, K.E., 1992, March. School bus routing using genetic algorithms. In Aerospace Sensing (pp. 387-398). International Society for Optics and Photonics.
[89] Tillman, F.A., 1969. The multiple terminal delivery problem with probabilistic demands. Transportation Science, 3(3), pp.192-204.

[90] Verderber, W.J., 1974. Automated pupil transportation. Computers and Operations Research 1 (2), 235–245.
[91] Wren, A. and Holliday, A., 1972. Computer scheduling of vehicles from one or more depots to a number of delivery points. Journal of the Operational Research Society, 23(3), pp.333-344.
[92] Xu, H., et al., 2003. Solving a practical pickup and delivery problem. Transportation Science, 37(3), pp.347-364.
[93] Yan, S., Hsiao, F.Y. and Chen, Y.C., 2015. Inter-school bus scheduling under stochastic travel times. Networks and Spatial Economics, 15(4), pp.1049-1074.
[94] Yao, B., et al., 2016. A two-stage heuristic algorithm for the school bus routing problem with mixed load plan. Transportation Letters, 8(4), pp.205-219.